Universidade de Aveiro
2013

Departamento de Eletrónica, Telecomunicações e Informática

André Gomes Barata

Sistema de Gestão de Iluminação em Ambientes Industriais

Lighting Management System in Industrial Environments

Universidade de Aveiro
2013

Departamento de Eletrónica, Telecomunicações e Informática

André Gomes Barata

Sistema de Gestão de Iluminação em Ambientes Industriais

Lighting Management System in Industrial Environments

Dissertation submitted to the University of Aveiro as part of its Master's in Electronics and Telecommunications Engineering Degree. The work was carried out under the scientific supervision of Doctor Luís Filipe Mesquita Nero Moreira Alves, Assistant Professor of the Department of Electronics, Telecommunications and Informatics of the University of Aveiro and Engineer António Manuel Rodrigues Tavares of Diferencial - Eletrotécnica Geral, Lda.

o júri

Presidente

Professor Doutor Alexandre Manuel Moutela Nunes da Mota
Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

Professora Doutora Mónica Jorge Carvalho Figueiredo
Professor Adjunto da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria

Mestre António Manuel Rodrigues Tavares
Especialista e Diretor da Unidade de Projetos de Investigação e Desenvolvimento da Empresa "Diferencial - Eletrotécnica Geral, Lda."

Professor Doutor Luís Filipe Mesquita Nero Moreira Alves
Professor Auxiliar do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

Acknowledgments

I wish to express my most sincere appreciation to my thesis supervisor, Doctor Professor Luís Mesquita Nero Moreira Alves, for the guidance, encouragement and critics.

I would like to spend a special thank to the company Diferencial – Electrotécnica Geral, Lda. for sponsoring and supporting this thesis and other university projects. I would like to address my gratitude to António Manuel Rodrigues Tavares, Diferencial's director of the project's unit, for the supervision and support during my dissertation.

Although there were many people who helped me during my dissertation, I would like to highlight and thank to Nuno Rafael Mendonça Lourenço. He made me feel at home during my brief stay at the Telecommunications Institute, I wish him all the best for his future projects and personal life.

I would also like to thank to my all times friends and to the ones I met during my University time. It was a pleasure to share all those moments.

I could not forget to thank to my mother Teresa Gomes, father José Barata and sister Ana Barata who were always supportive and where I needed them to be. I'm grateful for giving me the opportunity to enjoy and live such a dreamful life.

palavras-chave

Iluminação, sistema inteligente, controlo automático, economia de energia, domótica, inovação.

resumo

Esta dissertação aborda a implementação de um sistema de gestão de iluminação especialmente concebido para operar em ambientes industriais, assumindo-se como uma solução respeitadora do ambiente e preocupada com o conforto global de todos os seus funcionários.

O sistema foi construído com uma topologia mestre/ escravo, a fim de dar ao utilizador a necessária interatividade. Tem a possibilidade de ser utilizado em modo autónomo, aumentando assim a versatilidade do produto criado. Deste modo, pretende constituir-se como uma alternativa válida aos sistemas que estão no mercado na execução de novas instalações e poder ainda ser utilizado na substituição de instalações existentes sem que isso signifique um esforço financeiro desproporcionado. O sistema concebido tem um retorno de investimento num prazo muitíssimo curto.

Este trabalho aborda também a importância da utilização em ambientes industriais de lâmpadas LED, nomeadamente das novas lâmpadas LED de alta potência, que permitem o desenvolvimento de soluções muito mais eficientes e a emergência de novos conceitos de controlo e gestão de sistemas de iluminação. Estudos recentes dão conta do extraordinária evolução das tecnologias relacionadas com esta temática e as projecções que é possível realizar para o futuro apontam para a sua utilização em larga escala.

Sendo essencialmente um trabalho de investigação teórica, o mesmo foi enriquecido com uma componente prática que apoia as conclusões apresentadas.

keywords

Lighting, intelligent system, automatic control, energy saving, home automation, innovation.

abstract

This dissertation addresses the implementation of a lighting management system especially designed to operate in industrial environments. assuming a solution as environmentally friendly and concerned about the overall comfort of all its employees.

The system was built with a master /slave topology, in order to give the user enough interactivity. Has the possibility to be used in standalone mode, thereby increasing the versatility of the created product. This is intended to constitute itself as a valid alternative to systems that are on the market in the implementation of new facilities and may still be used as un upgrade of existing facilities without meaning a disproportionate financial burden. The designed system has an investment return.

This work also addresses the importance of utilization in industrial environments of LED lamps, including the new high power LED lamps, which allow the development of more efficient solutions and the emergence of new concepts of lighting management systems. Recent studies realize the extraordinary evolution of technologies related to this theme and the projections that we can do for the future point to their widespread use.

Being essentially a work of theoretical investigation, it was enriched with a practical component that supports the conclusions presented.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

ACK	A flag used in the Transmission Protocol to *ack*nowledge receipt of a packet
ADC	Analog-to-Digital Converter
ALC	Advanced Lighting Control
CAN	Controller Area Network
CTL	Control (Key)
DALI	Digital Addressable Lighting Interface
DIP	Dual In-line Package
DLM	Digital Lighting Management
DMX	Digital Multiplex
DST	Destination
EEPROM	Electrically Erasable Programmable Read-Only Memory
EHF	Extra High Frequency
EIA	Electronic Industries Association
EIB	European Installation Bus
FSK	Frequency-Shift Keying
HDR	Header
HF	High Frequency
HID	High-Intensity Discharge
HVAC	Heating, Ventilation and Air-Conditioning
ID	Identifier
IEA	International Energy Agency
IEC	International Electrotechnical Commission
KNX	Konnex
LDR	Light Dependent Resistor
LED	Light-Emitting Diode
LF	Low Frequency
MAC	Medium Access Control
MF	Medium Frequency
PCB	Printed Circuit Board
PHY	Physical (layer)
PIR	Passive Infrared Sensor
PLC	Power Line Communication
PWM	Pulse-Width Modulation
RF	Radio Frequency
RJ	Registered Jack (Ex.: RJ45)
RS	RS (Ex.: RS-232)
SAD	Seasonal Affective Disorder
SBS	Sick Building Syndrome
SCS	Systems Communications Services
SHF	Super High Frequency
SPST	Single Pole, Single Throw

SRAM	Static Random Access Memory
SWOT	Strengths, Weaknesses, Opportunities, and Threats
TIA	Telecommunications Industry Association
UHF	Ultra High Frequency
VHF	Very High Frequency
VLF	Very Low Frequency
XLR	Kind of connector found on professional audio, video and stage lighting equipment

CHAPTER 1:
INTRODUCTION

We are facing a huge change in the lighting industry, although the incandescent light bulbs and the halogen lamps are still the industry standards, the white LED (light emitting diode) solutions will take 20% of the world market in 2012 [1]. The usage of LED based solutions is known to be more energy efficient since they do not waste most of the energy as heat, and most of all, the life time of a LED based lamp is about 10 times higher than the incandescent light lamp. The only drawback is the higher initial cost to replace the traditional lamps to the LED based ones. This drawback is actually a new challenge for the scientific community which is forced to develop and improve lighting management systems to improve energy savings. Therefore, reducing the investment return period, make use of this technology full potential.

LED is a component capable of being switched at high rate, making possible to dim a lamp according to the user needs, characteristics which are not shared with the traditional lamps or, at least, in an efficient way. High-Intensity Discharge (HID) family of lamps needs to warm-up in order to work properly. In the startup period, they need to achieve the right temperature and pressure and that is the reason why it is so hard to use traditional PWM dimming process to control the illumination level. It means, to dim HID lamps would demand to change the typical operating conditions lowering even more the overall efficiency ratio.

Lighting is an extremely important factor when it comes to design an industrial facility. Recent studies have shown us what we can easily sense through our daily experience that lack of illumination drive us into state of Sick Building Syndrome (SBS) which includes headaches, lethargy, irritability and poor concentration. In addition to SBS, there is another syndrome called Seasonal Affective Disorder (SAD) whose manifestations are visible when experiencing walking difficulties in the morning, morning sickness, oversleeping and lack of energy and concentration [2]. Apart of SBS and SAD, which take only effect after long periods of exposure in low lighting environments, some industrial environments can be very hazardous. For example, a wood lumber company where workers are constantly operating heavy machinery, lighting must be a major concern when it comes to safety. A slight failure in the lighting system can induce the chaos at the workplace. Reliable systems that can quickly adjust to existing needs of better lighting must be used in such cases. All in all, a poor lighting system can be responsible for accidents and injuries, reduced working efficiency, productivity and overall comfort.

Among the qualities stated before, it is worth to highlight that LED based lamps can be classified as a clean technology, due to its mercury, along with other toxic metal, free constitution. Incandescent bulbs are known to be 75% less efficient than LED ones so, converting the existing lighting systems, makes a huge contribution towards the reduction of energy demands and increases the workers safety and comfort.

1.1. Solid-State Lighting

According to the International Energy Agency's study (IEA), 20% of United States generated electricity is consumed producing lighting. A closer percentage is also spent in Europe and this value can get higher in some developing countries. IEA has estimated that approximately 1900 Mt of CO_2 are emitted per year to the atmosphere. It is appropriate to refer that 80% of this emissions come directly from the electricity generation process [3].

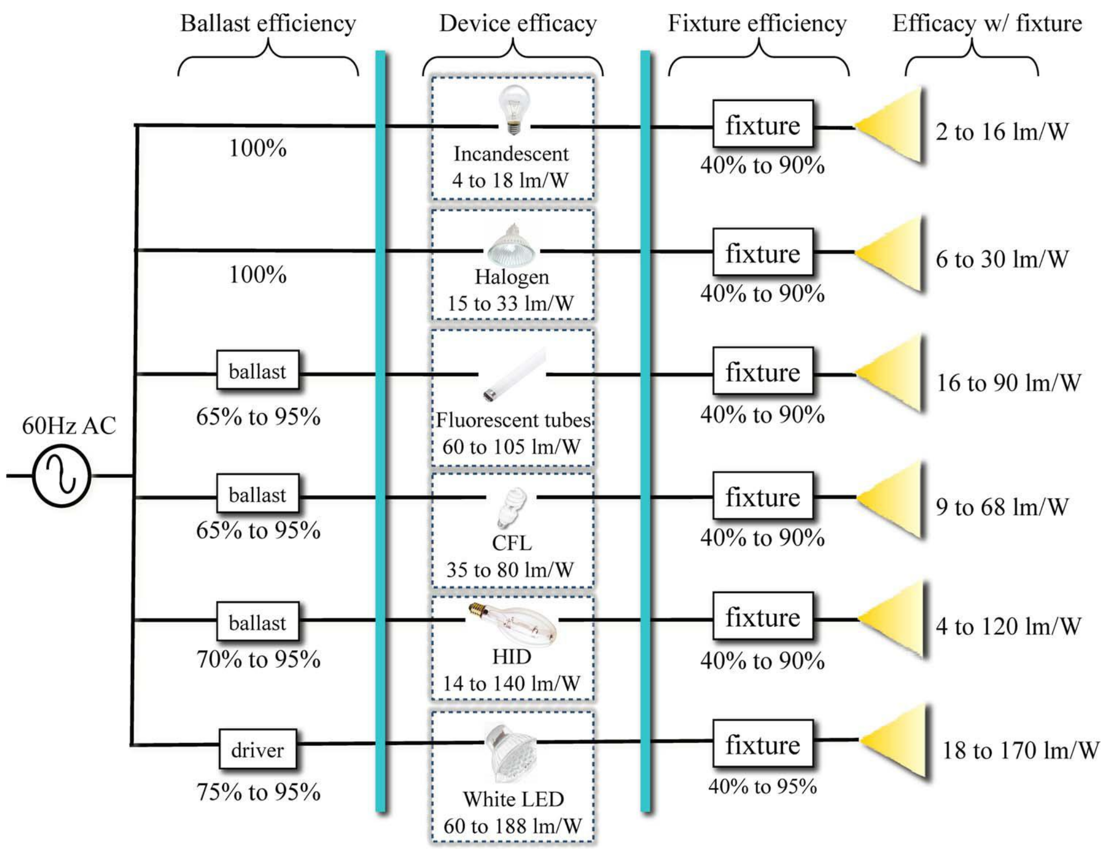


Figure 1: Efficacy of Lighting Devices and Fixtures [4]-[5].

The urgent need to replace the traditional lighting systems is not only a matter of reducing energy waste but also to contribute to a cleaner environment. Unfortunately we are living in a world where lighting efficiency is not the primary concern of consumers when it comes to pick their lighting solution. Nowadays consumers care more about factors such as cost and environmental pollution.

To clarify the global lighting efficiency panorama it was included an expressive image (fig. 1). In it is possible to realize that white LED is the most efficient lamp when compared, for example, with the incandescent bulbs, which convert only between 1% and 5% of the energy they consume into usable light [6]. The data in figure 1 date from 2009 and, in the meantime, all types of lamps improved their efficiency. The graph in figure 2 shows the cost evolution and luminous efficiency of LED packages at 35 A/cm^2. The cost of LED lamps will drop from 10 to 1 in a decade and regarding to luminous efficiency it is expected to reach values approximately 220 lm/W in 2020 [7].

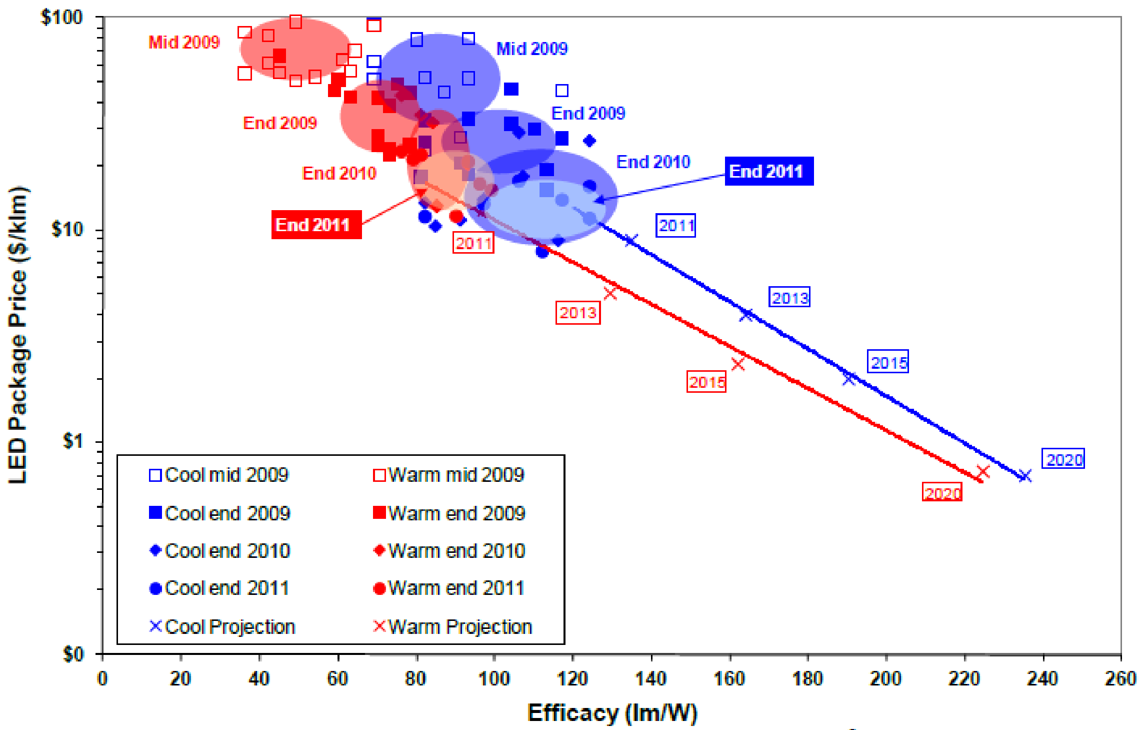


Figure 2: Price-Efficacy Tradeoff LED Packages at 35 A/cm^2 [7]

1.2. Objectives of the Thesis

The main goal of this thesis is to create management system capable of operating in industrial environments using high nominal power luminaries. As was stated before, nowadays industries use manly gas-discharge lamps but, since LED based lamps are achieving similar efficiency results, if not better than the currently industry standards, this new technology is a viable alternative and competitor. The desirable implementation is supposed to use this new alternative; therefore, some considerations have to be made regarding its use. Traditional lamps are not adequate when it comes to vary its luminous flux, because of their slow start. Characteristics not shared by LED, whose switching capability is possible at higher rates, making viable to adjust the luminous flux.

In the previous publication of the University of Aveiro master program in the field of Electronics, a similar project was developed – a lighting management control system capable of diming metal halide lamps. The present dissertation can be seen as an adaptation of the same philosophy but for the more modern high intensity LED lamps [8].

The desirable system has to be able to monitor the current light level and, judging by the sampled value, it has to adjust the illumination flux in order to achieve and meet the desirable threshold. It is imperative that this system implements strategies to reduce the overall consumption.

It is important to highlight that, one of the main goals of this project, is to comply with the possibility to be integrated in already existing facilities, in order to be an acceptable replacement to the existing lighting systems. This is considered to be a key on this project success.

1.3. Structure of the Thesis

It was decided to divide this dissertation into five different chapters, in order to give the reader an entire perspective of the project development process and also a good insight of what is currently in the market and the LED's state of the art. **(Chapter I** is the present section.)

Chapter II – gives a global overview of what lighting management solutions are made available in the market. The main focus of this chapter is to give the reader an overall view of what kind of protocols, sensors and physical media could be used in such lighting systems.

Chapter III – presents a fully detailed description of the overall developed system. Since this system was designed to operate in all possible industrial environments, every single implementation was explained and discussed to give the reader a full understanding of how it works and why it was chosen. It was highlighted the chosen system topology, its main constituent blocks and how the communication was implemented between devices.

Chapter IV – presents a fully detailed description of the developed system**.** The global functionalities of the project were explained as well as the implementation functional modes. In order to summarize all the pros and cons of the project, a SWOT analysis was performed. In the end of the chapter, a study cost and performance test was accomplished to give the reader a greater insight on the project.

Chapter V – in this last chapter will be discussed the results and conclusions of the dissertation. Although this project has come to an end, some guidelines and thoughts for future development works on this area were stated.

CHAPTER 2:
STATE OF THE ART

Prior to the conception of a brand new equipment or device it is indispensable to do a serious study about: what has already been done; what is, at the moment, available to the public; how was it conceived; what can be done to improve the existing solutions in the market or if it is possible to implement the same solution with a lower costing price. These questions are discussed on this chapter, always trying to explain it in a simple, yet effective way, focusing on the details that can be used for this work. Reflections about the existing solutions and protocols will be made with the objective of choosing/achieving the desirable and optimal solution for the given problem.

This chapter is structured in the following order: Physical Interfaces, presentation of the existing interfaces used to establish communication in industrial applications. Chosen solution will be present in the end of every sub-section; Communication Protocols, presentation of the most common communication protocols used in the domotic field; Sensors, brief study concerning the functional concept behind the existing sensors used in lighting management systems. In the Existing Market Solutions sub-chapter It was included an extensive study about the approaches done by the well-known market leader brands, along with a study cost.

2.1. Physical Interfaces

In order to establish communication between devices, actuators and sensors, a medium has to be used. It is important to highlight the standards of physical interfaces, used for industrial applications, should meet. Industrial environments are generally known to be crowed, partitioned, noisy areas. As a matter of fact, these are environments were a fault or a faulty system could lead to a chaotic situation so, if the aim is to build a lighting management system, the medium chosen has to be reliable and fault-proof, in order to prevent dangerous situations caused by a light failure.

Generally speaking, physical interfaces can be divided in two different groups: wired solutions and wireless ones. Both can be reliable, depending on the application or situations where they are used.

2.1.1. Wired Solutions

Within wired solutions there exist two different possibilities:

- dedicated – where the wires are used just for a specific purpose.
- non-dedicated – makes use of the existing wiring (power lines) to transmit the system data.

2.1.1.1. Power Line Communication

It belongs to the non-dedicated wired group. PLC, as it also known, is nowadays a well-matured technology and makes use of the already existing electrical installation to transmit data along the power lines. The concept behind PLC has started being developed in the 50's. How can this be done? Transmitters modulate data into a carrier wave, which is then injected directly into the power line. Receivers, in its turn, de-modulate the received data for further use (fig. 3). Since PLC makes use of the existing wiring and considering that every device is powered, this means, theoretically, all powered devices can be controlled and monitored without any extra installation cost.

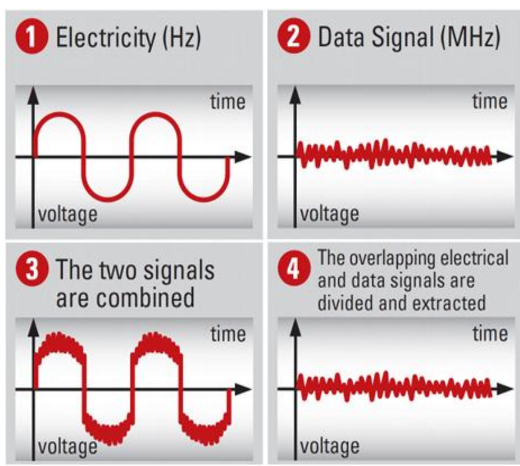


Figure 3: PLC Transmission

PLC can be used in numerous ways and for completely different types of applications, it is even possible to transmit internet. In the figure 4 we have an example of a PLC implementation. It is expectable that it will take a major place and become the number one solution in public light management systems [9] (as we can see one example in the figure

4) but for industrial and home purpose it does not look so attractive. As it is known, industrial environment has huge number of inductive loads such as heavy machinery and induction motors which can bring interference to the communication process.

In the past, Legrand has adopted PLC based solutions but, in December 2009, decided to discontinue their product line due to the changing marketplace and the desire to improve the products offered to the customers.

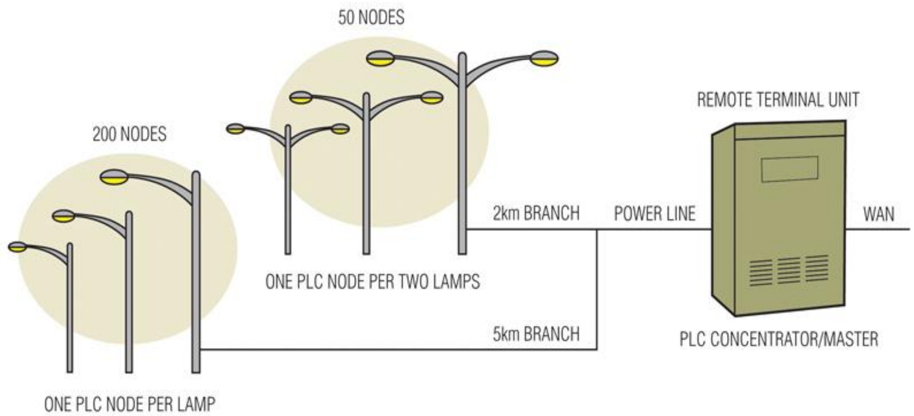


Figure 4: Street Lighting Network Using PLC to Communicate.

2.1.1.2. RS-485

RS-485, commonly referred as TIA/EIA-485, is a popular standard in fieldbus systems. This standard characterizes the specifications of drivers and receivers to transmit a balanced signal over a long distance in a noisy environment. In the 80's Electronic Industries Association (EIA) and the Telecommunications Industry Association (TIA) joined forces to develop this standard and, from that day on, it has been largely implemented in numerous ways and in different fields of application [10].

This standard is the logical successor of RS-232, which has been around since 1969 and it is still used in applications such as programming microcontrollers. RS-485 adds some improvements to its predecessors like the unique feature to operate with multi-transmitters or increasing the drive capacity. The main improvement is definitely the extended range of operation, although this is directly associated to the transmission rate. The longer is the line between equipments/transceivers the slower will be its connection. RS-485 provides the physical link for data exchange, enabling the transmission of serial messages using a multipoint bus - multiple transceivers, drivers or receivers per bus line. The communication frame, scheme for addressing and collision management, among other tasks, has to be defined by the communication protocol.

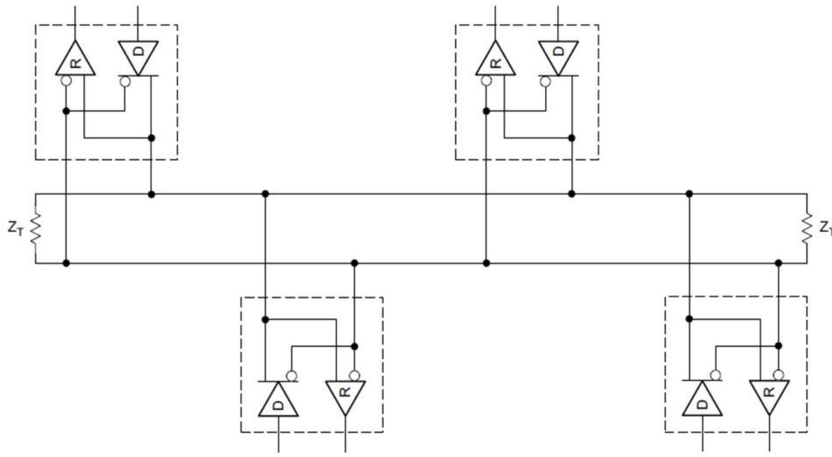


Figure 5: Daisy Chain Arrangement Using RS-485 [10].

RS-485 was designed to operate both in half-duplex and full duplex mode. Considering the possibility to run a system communicating in full-duplex mode, two transceivers and four cables will be needed. Figure 5 represents a RS-485 bus running in half-duplex mode.

As it was stated before, with RS-485 it is possible to operate in multi-master configuration or master-slave mode. In any chosen method, some rules have to be followed in order to assure a safe communication like solely having a master assigned at the same time, which means only master(s) can initiate the transmission process. When running in multi-master mode, a slave node that wants to become master has to notify the current master before transiting between states. The most common applications run in master-slave mode which means that the system will have a master and multiple slave nodes. Since RS-485 only integrates the physical layer, this kind of decisions have to be contemplated on the communication protocol.

Although the transmission rate is not really an issue when projecting a lighting management system, since the data collected by the sensors does not require high bandwidth, the distance between controlling board can be, in some cases, significant. In the table 1, it can be seen the relationship between speed rate of transmission and distance. With RS-485 it is possible to communicate at 100 kbit/s using 1200 meter of twisted wiring between equipment.

Distance (m)	Speed Rate
10	35 Mbit/s
50	2 Mbit/s
1200	100 kbit/s

Table 1: RS-485 Speed Rate Table. [11]

2.1.2. Wireless Solutions

The concept behind wireless communication is very old since Humanity began to use smoke signals and fire to transmit information mainly in case of war. Many years have passed from that date and, nowadays, imaging the world without wireless communication is almost impossible. Since wireless communication uses air as its propagation medium is no longer needed to waste money in cooper cable. With less restrictions in the connection possibilities, it is predictable that the number of applications, using wireless connection, will grow in the future. Although wireless connections give a lot of possibilities, they all have the same drawbacks:

- surrounding noise
- influenced by geometric characteristics of the environment

2.1.2.1. RF

RF is the acronym for radio frequency, which is known to be the frequency of radio waves - one form of electromagnetic radiation in the range of 3 kHz to 300 GHz.

This section of electromagnetic spectrum is located under the frequency of the visible light and the infra-red zone. Although different names can be given to different sections of radio frequency spectrum, the only difference is their wavelength / frequency (fig. 6). They are named differently to indicate a specific region and use purpose. The main domotic RF protocols use the legislated frequencies such as: 436 MHz and 838 MHz. It means they are located in the Ultra High Frequency range [12].

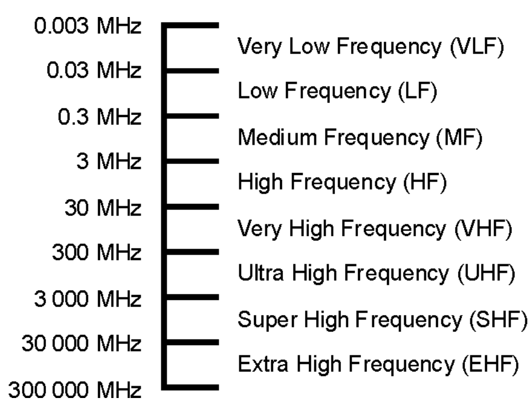


Figure 6: Wireless Frequency Spectrum.

2.1.3. Final Comparison

In order to wrap around all the characteristics and differences between those two implementations, a comparison table was made (tab. 2).

Final Comparison			
Features:	**Wireless**	**Wired**	**Source**
Transceiver price	€9,96	€1,16	http://pt.farnell.com
Range (meter)	> 150	1200	
Advantages	• No cables • Flexibility	• More reliable • Faster speed rate	
Cons	• More expensive • Slower speed rates	• Implementation cost • Installation	

Table 2: Communications Final Comparison

In the previous table it is shown a final comparison between wired and wireless implementations. The chosen transceivers were the Hope RFM12 module (wireless) and MAX487CPA (wired), one RS-485 transceiver. The difference between transceiver cost is relevant but it is important do not forget about the installation cost that wired implementation requires. Since it is required to install a point-to-point wired connection, the global wired implementation cost can exceed the wireless cost. Along with the implementation being less expensive, wireless can be used in a huge variety of scenarios. The most relevant drawbacks in the wireless implementation are: the operation range where they can operate without noise interference and the communication speed rate.

2.2. Communication Protocols

In order to create a communication link between devices, a common protocol is needed, which must be able to agree on:

- the data format which is being transmitted.
- the transmission
- the priority
- and whether the transmission is synchronous or not

In the next subsections are described some important communication protocols used in the domotic and lighting field.

2.2.1. Konnex (KNX)

KNX protocol is based on EIB protocol and condensates some other two existing protocols - European Home Systems Protocol (EHS) and BatiBUS.

EIB (European Installation Bus) protocol was developed by a group of companies, leaders in the European electronic material market. Their main objective was to put a stop to the importation of American and Japanese solutions which were, in many ways, superior to the ones built in Europe. The concept behind, was the creation of one European norm allowing the intercommunication between devices installed in a network, no matter it was a house or an entire building. EIB was designed to operate as a decentralized architecture, granting individual intelligence to every sensor or actuator.

KNX was intended to: create a standard in the domotic and automation field which fulfilled the European requisites; introduce the Plug & Play principle in every common house dispositive; overcome the previous problem, concerning efficiency improvements on the communication process.

Key features (tab. 3):

Twisted Pair	Wireless (KNX-RF)
Maximum segment length – 1000 m	Frequency of operation – 868 MHz
Data rate – 9.8 kbps	Data rate – 16 kbps
Maximum 57600 network nodes	Max. range: 800 m (outdoor)
Devices within same physical segment are addressed with 8-bits	

Table 3: Key features of KNX protocol [13], [14]

Configuration modes: Konnex defines not only different media, but also different configuration modes. In principle any combinations between a configuration mode and a medium is valid. There are three categories of KNX device:

- A-mode or "Automatic mode" devices automatically configure themselves, and are intended to be sold to and installed by the end user.
- E-mode or "Easy mode" devices require basic training to install. Their behaviour is pre-programmed but has configuration parameters that need to be customized to the user's requirements. (For KNX-RF only the easy mode is used.)

- S-mode or "System mode" devices are used in the creation of custom-made building automation systems. S-mode devices have no default behaviour and because of the complexity of the system mode, it is foreseen for professional installers [14].

KNX protocol is not only a designation or a way of encoding the transmitted data. It is a label who grants consumers not only the support of the protocol but also fully compatibility with all the labeled KNX products connected to the bus.

2.2.2. X10

X10 was the first protocol, created in 1975 by Pico Electronics, aiming to fulfill the needs of the domotic world and home automation. Although the years have passed it still goes on one of the most used protocols. X10 Ltd, the company that bought Pico Electronics, was the first using PLC technology to control light, heat and security systems. They were even responsible for making PLC available on open market.

X10 protocol can address up to 256 different addresses. The message transmitted starts specifying which house is this message for (16 different house codes); followed by the unit code (16 different unit codes) and a simple encoded command such as: "All Units Off"; "Status Request" or "Dim". These encoded commands are already established in the protocol which will make it "simpler and limited".

2.2.3. Digital Addressable Lighting Interface (DALI)

DALI is not only a communication protocol, it also includes the electrical interface, and they were both defined by the standard IEC 60929/EN 60929. DALI is an international standard for lighting control systems providing the required interface to monitor and control dimmable ballasts, relay modules and controllers. Although it was initially designed to operate with ballast lighting, now it is also available to operate with LED based solutions.

In a brief approach, DALI requires a single pair of twisted wires to communicate with all the devices within the same network. The transmitted data is encoded in manchester code

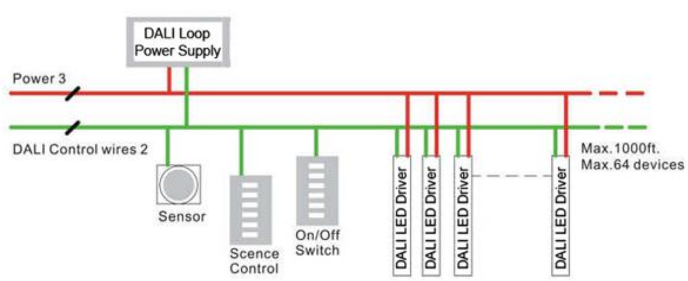


Figure 7: DALI Network Implementation

in order to grant a high signal to noise ratio which will enable a reliable communication between devices. Signal level is defined as 0 ± 4.5 V for "0" and 16 ± 6.5 V for "1".

DALI, as a standalone solution, can address up to 64 devices and, when used in a subsystem configuration, it can address more than 64. Data is transmitted at 1200 bit/s speed rate in a half-duplex mode. DALI messages consist of an address part and a command part. The address part determines which DALI module the message is intended for. All the modules execute commands with "broadcast" addresses. 64 unique addresses are available plus 16 group addresses (fig. 7) [15].

2.2.4. DMX512

DMX512 is a simple digital protocol widely accepted in the stage lighting industry. The concept behind its implementation is to create a light management network like DALI. This network is composed by one DMX master node and a maximum of 512 other secondary nodes such as actuators or LED modules. Since this protocol was conceived to operate for speed and simplicity it lacks error checking functionalities. Therefore, it is not recommended for hazardous applications.

The physical layer is implemented based on the EIA-485 standard. The interconnection between devices is done by a multi-drop bus whose connections are ended up with XLR connectors. Although this system was originally designed for XLR connectors, it also supports RJ-45 connectors.

Generally speaking, every node will be in "listening" mode, waiting the income of commands. This protocol is not command based, like X10 or DALI. What is transmitted is the brightness value [16]. Figure 8 shows the DMX512 transmission frame.

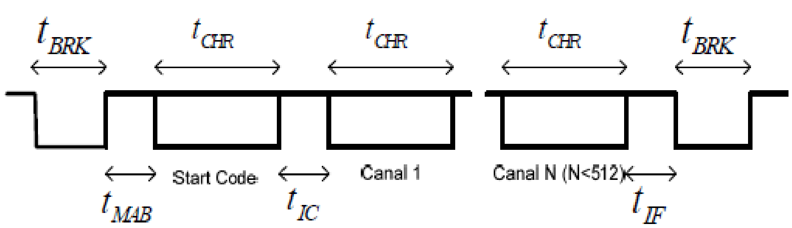


Figure 8: DMX512 Frame [16]

2.2.5. CANopen

CANopen is a communication protocol, at the beginning for field bus type CAN (Controller Area Network) running in real time.

Others include recently CANopen bus as (EtherCAT, Powerlink) thus demonstrating the interest of the industry for this type of communication. It is used in many fields: automotive, agricultural, industrial (elevators, escalators, motion control) and medical (X-ray, operating theaters). The fieldbus is known to be a solution to economic and effective communication.

Key features:

- Data rates of approximately 1 Mbps
- Up to 40 meters range
- Can handle communication collisions (CSMA/NBA)

2.2.6. MiWi

MiWi is proprietary wireless protocol designed by Microchip Technology. This protocol was designed to be used in RF transceivers based on the IEEE 802.15.4 standard for wireless personal area networks (WPANs). The most relevant specifications are: low data transmission rates, short distance transmission and low power consumption. They are ideal to be used in industrial monitoring and controlling devices, home and building automation and low-power wireless sensors. [17]

2.2.7. Hope RFM12B

Since its foundation in 1998, Hope Microelectronics Ltd has become one of the most successful designers and manufacturers in digital sensor, RF IC, RF discrete devices and related application solutions. Hope's products have been widely accepted by reputable global companies and extensively being used in diverse fields such as: wireless sensor, domotic applications and wireless high-speed data acquisition system.

The RFM12B transceiver is: a single chip designed to be used in low power applications; a multichannel FSK transceiver designed to be used in unlicensed 433, 868 and 915 MHz bands. This transceiver is a flexible and low cost alternative to a ZigBee. Its protocol: support mesh implementation; has acknowledge system and does not require licence [18].

2.2.8. ZigBee

ZigBee is a network protocol created to standardize the operation mode of 802.15.4 Wireless Networks in industrial environment. It works over the PHY and MAC layers of the 802.15.4 standard and adds functionalities such as Routing Capabilities, improvements in Security and other general improvements, concerning energy efficiency and communications reliability.

Key features:

- Data rates of approximately 250 kbps

- Up to 100 meters range
- Fully hand-shaked protocol for transfer reliability

ZigBee was designed for domotic networks. It appeared as response to market demands. This implementation aims to fulfill the needs of a wireless transceiver able to send, in a reliable way, small packages of information at a low cost [19].

2.2.9. Final Comparison

In the table 4, the comparison of the features of the protocols.

Final Comparison				
Protocol	**License Required**	**Media**	**Max. Transmission Rate**	**Acknowledge System**
KNX	✓	Both	9.6 kbit/s	✓
X10	X	Both	20 bit/s	x
DALI	✓	Both	1.2 kbit/s	✓
DMX512	X	Both	250 kbit/s	x
CANopen	X	Both	1 Mbit/s	✓
MiWi	X	Wireless	250 kbit/s	✓
RFM12B	X	Wireless	115.2 kbit/s	✓
ZigBee	✓	Wireless	250 kbit/s	✓

Table 4: Protocols Final Comparison

As it was stated in the objectives of this dissertation, this project goal is to create a brand new lighting management system capable of operating in industrial environment at a reduced price. It means that the desirable solution should be, if possible, based on non-licensed protocols. Possible choices rely on: CANopen, DMX512 or X10. In the physical interface section it was said that PLC based solutions are getting out fashion and, adding this to its speed transfer rate, X10 seems not to be a reliable solution/protocol. Which leave us with CANopen and DMX512. All in all, CANopen seems to be the most reliable protocol of these two but the reason why it was chosen DMX512 is due to the fact that is the only one completely related with lighting systems. If the goal was to create an implementation capable of being improved, changed by a third party, the most reasonable choice is to go for a fully dedicated lighting management protocol such as DMX512. Using ZigBee was impracticable due to the need to pay a commercialization fee, therefore the decision relied on choosing between RFM12B and MiWi, for the wireless implementation. During the decision process it was analysed how difficult would be implementing such protocols in the future developing platform and since it was not chosen a Microchip microcontroller, implementing a Microchip property protocol would be impracticable. Since MiWi had no major advantage over RFM12B, the last was chosen one for the wireless communication.

2.3. Sensors

Consulting some English dictionaries, the word sensor is described in some different ways such as: "anything that receives a signal or stimulus and responds to it" [20]; "a device which detects or measures a physical property and records, indicates, or otherwise responds to it" [21]. Sensors are so common and essential in almost every field of electronics, from medical to lighting industry, and very little times we know what are the physical working phenomena behind it. In this subsection, and regarding only the adequate sensor for lighting management system – occupancy, motion and photosensors – are going to be discussed.

2.3.1. Occupancy Sensors

Ultrasonic Occupancy Sensors use ultrasonic diffusion technology to achieve 360° occupancy sensing. This is achieved by converting electrical energy into ultrasound waves in a frequency above the range of human hearing and it can even sense motion in areas with partial obstructions, through the principles of the Doppler’s effect. The frequency analysis of the received echoes will determine if the room is occupied or not. It is ideal to use this kind of sensors in: open spaces, large areas and areas unaffected by obstacles. Multiple sensors may be used to control large partitioned office spaces.

2.3.2. Motion Sensors

In a very brief way, the Passive Infrared Sensor (PIR) detects the occupancy using the heat difference between human being and the surrounding space. This solution was designed for small, enclosed spaces with frequent movement. In order to work, this kind of sensor has to be properly placed since it needs to have a full view of the compartment. It means that these sensors require, to work properly, a non-obstructed line of sight between the warm body and the sensor. It is imperative to take close attention in the moment of placing the sensor, since it is more sensitive, thus and more likely to detect the motion when placed perpendicularly to the trajectory.

PIR sensors are made of pyroelectric materials. The trick behind the detection is adding a Fresnel lens. These lenses divide the visible space into separate zones. When a person crosses a zone to another, it will stimulate the sensor. After that, the signal is amplified and it is possible to embed this sensor in a system, such as a security or lighting system.

2.3.3. Light Sensors

The light sensors measure ambient light level and they are "the key element" when it comes to lighting management systems. Since the goal of these systems is to keep a constant light index in the monitored area, knowing the exact level is crucial to switch on/off or dim one designated zone.

All the types of light sensors work through the same principle which is the photoelectric effect. When this materials are exposed to light they change their properties, for example, the light dependent resistors (LDR) when directly exposed to light their resistor properties change accordantly. Others than LDR, a light sensor can be conceived through three different implementations: photodiodes, phototransistors and photocells.

2.4. Existing Market Solutions

There are available on the market several solutions for the development of intelligent management of lighting systems. In general, we can say that all traditional big brands have their own system. In this sub-chapter will be analysed two different lighting management solutions already existing in the market. These two approaches were conceived by the well-known market brands: Legrand and WattStopper. In the subsection 2.4.3., it will be presented a case study based on the Wattstopper solutions.

2.4.1. WattStopper

WattStopper is an American company founded by the Mix brothers. Jerry and Steve Mix had a great conviction about the importance of providing energy efficient, convenient and accessible controls for working environments. In the development of these beliefs, WattStopper team introduced an innovative occupancy sensor product line. Nowadays, they expanded its occupancy sensor product offer and started developing lighting management solutions.

Product description:

WattStopper has developed an intelligent, distributed control system that optimizes light energy efficiency. They claim that their system, due to its powerful features, provides a higher investment return than any other lighting solution.

Figure 9: Connecting Detail of WattStopper's Implementation [22]

Their solution is called Digital Lighting Management (DLM) and it was designed to operate as a standalone control of individual building spaces, for a centralized control of a complete floor, building or even an entire University campus. DLM offers a vast number of solutions and control strategies to fit every possible scenario, giving the user the ultimate possibility to choose the ones that suits its interests. DLM convenience energy efficiency achieves the best results in spaces, such as: private offices, classrooms, common areas and conference rooms.

WattStopper developed a modular design to make installation easier: simple wiring diagrams, a single type of low voltage cables and not required high skilled installer. In the figure 9, a connecting detail.

The implementation was designed to use Cat 5e cables with pre-terminated RJ45 (fig. 10). This combination allows a quick, error free and accurate installation. Point-to-point connection and easy-wiring technology are the key features of this solution.

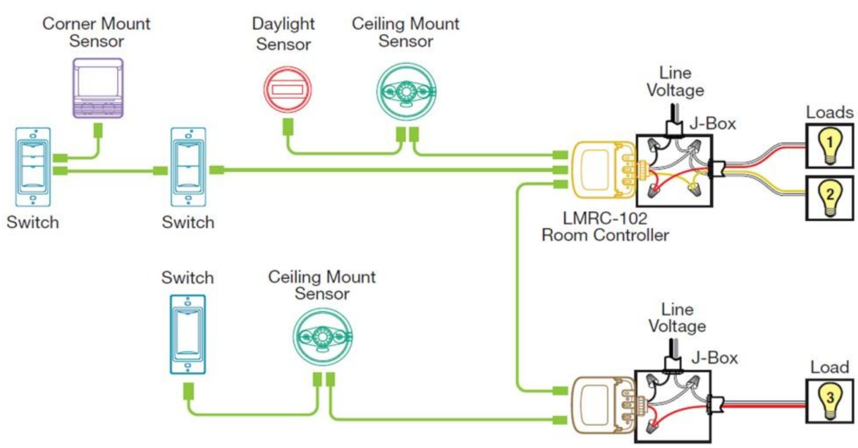


Figure 10: WattStopper's Concept Design [22]

Lighting Management Strategies

Occupancy-based control - Is the basic strategy in lighting management. The system is triggered when a presence is detected and it only goes off when it leaves the monitored area.

Integrated HVAC (Heating, Ventilation and Air-Conditioning) **control** - This is an upgrade of the previous feature. When a building space becomes unoccupied, non-essential plug loads such as task lighting, computer monitors and printers can be plugged into outlets that can be turned off automatically right after the sensor stops detecting presences. Considering presence lights, such as the ones embedded on the monitors or the consumption arising from idle status devices, they represent a huge parcel in the electricity bill. WattStopper found their way to reduce this energy wasted integrating HVAC control to their occupancy control.

Daylight-responsive control - A common feature, in nowadays lighting management systems, is the ability to compensate the light level when it is insufficient and WattStopper solution is not different.

Scene control and dimming / Personal controls - Offer occupants the possibility to easily change and adjust light levels for improved comfort and ambience. Research indicates that personal controls typically increase productivity and energy-efficiency because people feel more satisfied with their workplace and engaged in tasks longer [22].

2.4.2. Legrand

Legrand is one of the biggest industrial groups in France. It was founded in 1860 but only started to focus exclusively on the production of electrical wiring accessories for installations in the late 40's. Nowadays, Legrand is one of the global specialists in electrical and digital building infrastructures and is currently making research in the fields of sustainable development and energy saving.

Legrand has developed a system pretty similar to what is expected to be developed during this project. A system capable to provide the optimal lighting conditions in building/working spaces, such as: enclosed offices, conference rooms, classrooms, lunch rooms, libraries, rest rooms, hallways, and lobbies.

Solutions Available

Two types of solutions are available with Legrand's system. The first one, using only sensors directly connected to the luminary. These connections may be performed using wired solutions, such as RJ-45, or wireless ones, using a radio communication protocol like ZigBee (fig. 11). The second implementation is based on network connected through a bus (SCS system) which allows a more complex structure and control. This last solution is more suitable to monitor larger building spaces. Zigbee sensors can also be used if needed.

Figure 11: Legrand's System Assembling Options [23]

Lighting Management Strategies

Lighting management strategies refer to the method that can be used to control the existing lighting systems. Legrand's system is capable to operate and manage light in five different ways [20]:

Occupancy-based control – The system is triggered when a presence is detected in a determinate area and turns off when someone leaves the monitored area. In this mode, there is no possibility to control the light level therefore the system operates in ON/OFF mode. This is a common feature in nowadays lighting systems.

Vacancy-based control – The system is only triggered when someone pushes a button. After pushing the button, this mode operates like the occupancy-based control mode. At first, this strategy can seem a little redundant but Legrand claims that, adding this control, can provide an additional 10% energy saving above the significant saving already offered by presence detection.

Daylighting level control – This mode was developed to work in an environment where the luminary index may vary. This environment could be working spaces with light entrances. As we know, due to the day cycle, the light level will vary along the day and this will influence the amount of light required in one room. The system is triggered when a presence is detected and from that moment on, the light parameters are being constantly monitored. If the light level is lower that the predefined threshold, the system will compensate it until this threshold is met.

Scheduled control – Using the scheduling option, the user can set the system for a determinate period of time, to display the desirable luminary index. This feature could be very reliable for schools and offices where the schedule is already predefined.

Dimming control – This last feature allows the user to choose and change the brightness of the light.

It is possible to choose more than one working mode to function at the same time. Combinations such as:

- Scheduled control + Dimming control – can be very useful, when used at the same time, to create comfortable working ambience.

- Occupancy control + Daylighting level control – together they provide the optimal lighting management possible, stopping excessive energy waste.

2.4.3 Case Study

In this section, one case study will be presented in order to give a deeper perception of how the solutions work in a specified environment. This study will allow us to get conclusions about how much can be saved using WattStopper solution [22].

Scenario:

An open office with 160 m^2
Number of cubicles: 16
Area per cubicle: 10 m^2

Physical Representation (fig. 12)**:**

Figure 12: Case Study Physical Representation

Assumptions (tab. 4)**:**

Hours per year: 8760

Utility Rate: **€ 0.13/KWh**
Labour rate: **€ 70/h**

Annual building operating hours: 3146
General lighting power density: 11.8 Watt/m^2
Task lighting: 16 W/cubicle
Plug load power density (non-essential): 4.3 Watt/m^2

General Lighting:	**Task Lighting:**
Automatic-on/automatic-off: **30%**	Automatic-on/automatic-off: **30%**
Manual-on/automatic-off: **25% additional savings**	**Other Plug Loads:**
Bi-level automatic-on to 50%/Automatic-off: **34% additional savings**	Automatic-on/automatic-off: **75%**

Table 4: Assumptions [22]

Baseline Annual Operating Costs (tab. 5 and tab. 6)**:**

General Lighting:	**Other Plug Loads:**
11.8 Watt/ m^2 x 3148 h/yr x 160	4.3 Watt/ m^2 x 8760 h/yr x 160 m^2
Task Lighting:	
16 W/cubicle x 16 cubicles x 3148 h/yr	

Table 5: Baseline Annual Operating Costs [22]

Annual Operating Costs	Annual Energy Use	Annual Energy Cost
General Lighting	5943 KWh	€ 773
Task Lighting	806 KWh	€ 105
Other Plug Loads	6027 KWh	€ 784
Total	**12778 KWh**	**€ 1662**

Table 6: Annual Operating Costs [22]

Material needed for the installation (tab. 7)**:**

Unit	Cost	Misc. Material	Labour Hours	Nº	Installed Cost
DLM Occupancy Sensor	€ 145	€ 10	0.1 h	4	€ 620
Dual Relay Room Controller	€ 105	€ 50	0.5 h	1	€ 155
Two-button Wall Switch	€ 46	-	0.2 h	1	€ 46
Plug Load Room Controller	€ 53	-	0.1 h	4	€ 212
Total Installed Cost			1.5 h		**€ 1138**

Table 7: Material Needed for the Installation [22]

Plant :

As it is shown (fig. 13), the office is divided in four equal sections (4 cubicles each). One occupancy sensor per section detects the presence in the nearby area, allowing the system to operate separately. The system is only turned on when manually activated by the switch. After this moment, the light level is constantly monitored and compensated, if required [22].

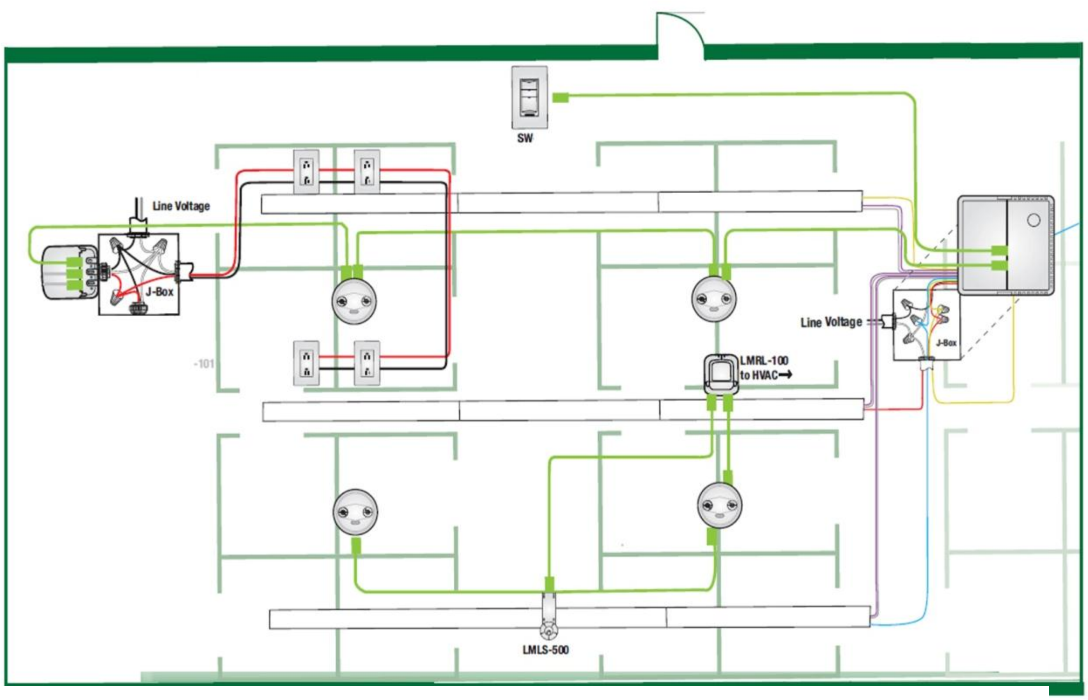


Figure 13: Partitioned Office Plant [19]

Investment Return:

Using the information and data gathered in WattStopper's website and brochures, it was possible to estimate the investment return. Based on the energy saving percentages shown previously, table 8 was built to point how much can be saved.

	Before Installation		% saved	After Installation	
Annual Operating Costs	**Annual Energy Use**	**Annual Energy Cost**		**Annual Energy Use**	**Annual Energy Cost**
General Lighting	5943 KWh	€ 773	64	2140 KWh	€ 278
Task Lighting	806 KWh	€ 105	30	564 KWh	€ 74
Other Plug Loads	6027 KWh	€ 784	75	1507 KWh	€ 196
Total	**12778 KWh**	**€ 1662**	**67**	**4211 KWh**	**€ 548**

Table 8: Overall Operating Costs [22]

Total installed cost: € 1138; total saved = € (1662 – 548) = € 1114. Respecting the assumptions made, this system will pay itself in one year and one month [22].

CHAPTER 3:
LIGHTING CONTROL SYSTEM - GENERAL SPECIFICATIONS

At this point, a good insight on the available solutions in the market has already been given. In the past chapter, it was studied the most relevant protocols concerning the domotic field, so the next step is to come up with the best implementation, covering as many applicable scenarios as possible. It is important that this system fulfils the desirable objectives, described in the first chapter of this dissertation, but, if possible, to introduce new features and applications. This chapter will give a general overview of the designed system. The concept will be explained along the chapter as well as the reasons which lead up to it.

The main goal of this system is to control and monitor an industrial lighting installation, fading up/down LED luminaries, in order to provide enough illumination to a determinate room space. Depending on the scenario, this lighting system has to be able to detect presence and trigger automatically, if someone enters the monitored area staying in the same state until presence is no longer detected.

In a previous publication of the University of Aveiro master program in the field of Electronics, was developed a lighting management control system capable of operating the metal halide lamps [8]. The chosen implementation had to be able to control lighting installation switching contactors on/off, in order to fulfill the illumination needs inside the factory hall. With this task in mind, it was chosen a master-slave topology where the control system (master board) had to be equipped with an optical sensor, control device, which will

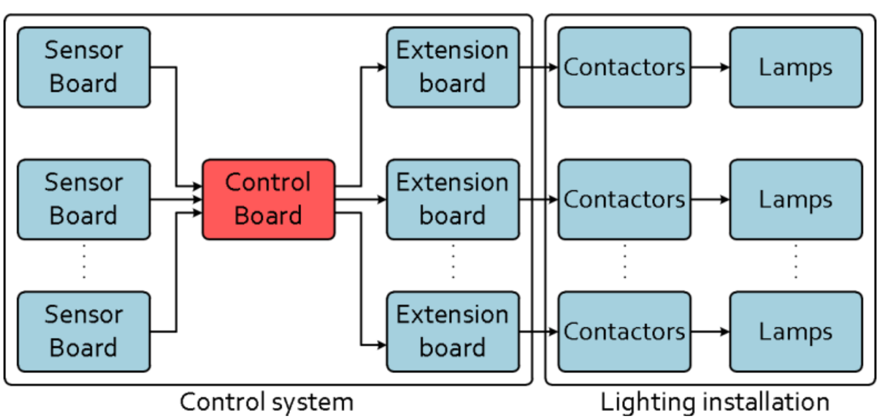


Figure 14: Control System Design [8]

manage all operations, and switches controlling the contactors. As it can be seen in the figure 14, the master board or control board would have as many sensor boards and extension boards attached as allocated zones. This implementation is granting the whole decision process to the master board, relegating the action procedures to the attached boards.

The present dissertation can be seen as an adaptation of the same design but for the more modern high intensity LED lamps. Although the strategy followed in the past proofed to be reliable, in this project, it was decided to keep the master-slave topology but granting the decision process to the slave boards. This decision was not taken lightly; according to the research presented in the state of the art chapter, nowadays lighting management systems tend to give decision-making power to the slave board or, at least, the decision process is commonly not centralized in one master board.

3.1. Master-Slave Topology

Before describing the chosen philosophy, it is necessary to briefly summarize the overall aspects of an industrial environment. An industrial installation is manly constituted by: large open spaces, which require different lighting profiles depending on the type of work performed, and individual or group offices. Each of these areas has its own specification, since offices are areas with greater human activity than open spaces. To enrich this implementation, it was decided to not only use a wired communication solution. Due to the possibility that some industrial environments may have an uncommon room displacement/arrangement this fact would not allow an easy implementation of a wired lighting management system. This

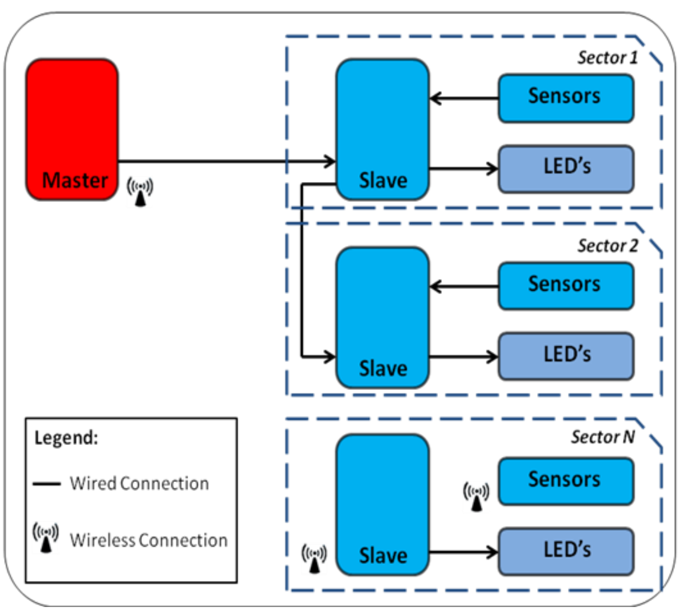


Figure 15: Proposed Design Implementation

system can transmit data through wired and wireless interfaces. Since the most relevant details have been stated, regarding the environment where the system will operate, it is time to describe the designed implementation.

In the figure 15 is shown the proposed master-slave topology designed for this project. The master (in red in the fig. 15) provides the user a full control of the illuminated areas. Each independent area has an allocated slave board where is attached a LED driver, a lightning sensor and if needed a PIR sensor. This last sensor is optional depending on which kind of area is the system being used. Every sector has its own dedicated sensor boards and drivers.

Although they are called slaves, they will play the main roll assuring the correct and desirable illumination. The master will work more as user interface of this system, since it will not have any algorithm control or information on sensor readings. The master board was designed to be implemented in an electrical distribution board and the slave ones are placed close to the allocated section. This master board will have the ability to override/adjust local definitions and this is the reason why it is called master board. With this topology less cabling will be required, since the sensors are just connected with its sector slave board and not with the master board itself. The design implementation in the figure 15 covers all the possible scenarios made available with this architecture. The main difference between sectors lies in the fact that one uses a wired connection and the other a wireless one.

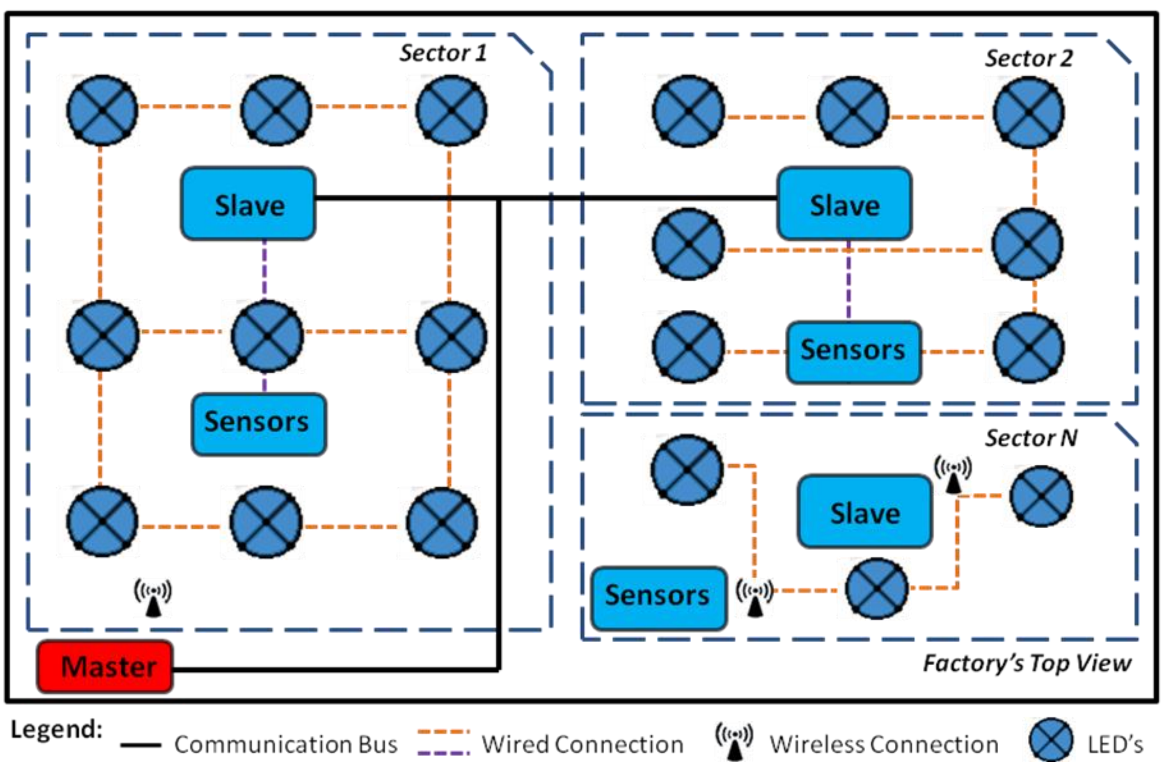


Figure 16: Lighting Management System Implemented on Factory

To clarify the whole concept of this system, it is shown a hypothetical practical implementation for this system, in the figure 16. In it, three different zones can be identified. According to the designed concept, every single one can have a different lighting profile.

The associated sensors are placed in the nearby, in order to sample a correct light value or to detect motion properly. Since wireless communication was used in the 'sector N', it is expected to be a zone where implementing any other solution would be pricy or impossible due to its displacement.

At this point, the overall philosophy behind this implementation was already explained. The next step will be describing in detail the constituents of master and slave blocks. In the figure 17, it is represented the connection between the master and slave (sector) as well as it is described the main component parts of this project. For a better understanding both will be explained in different sub-sections along with an explanation about the role they play in this implementation.

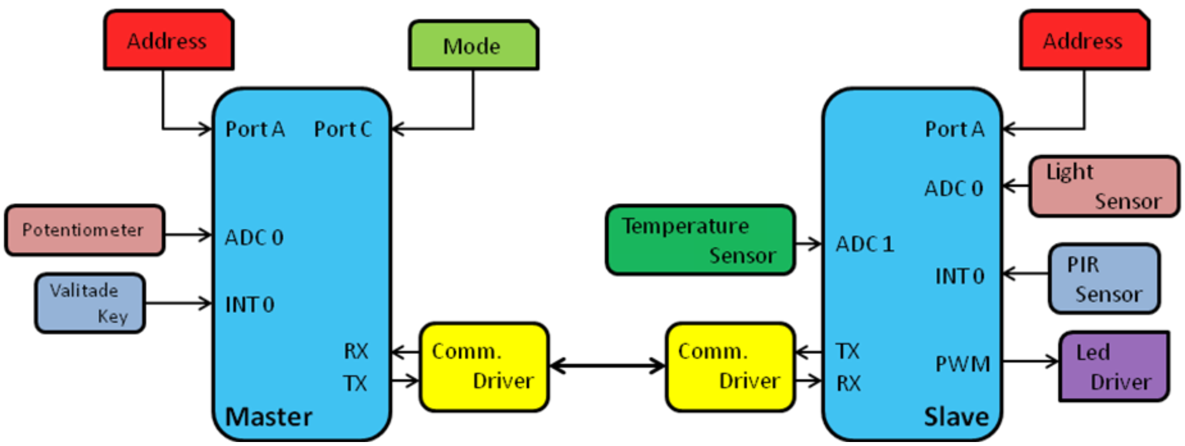


Figure 17: Master-Slave Block Diagram

3.1.1. Master

The master board, represented on the left side of the figure 17, is the interface between the user and the system. Its main function is to assign a determinate light profile to a certain addressable sector. Master board is constituted by five different blocks, which will be described in depth, in order to give the reader a global functional idea of the project.

Address block - It is constituted by a dual in-line package, from now on referred as DIP, connecting directly to the input port of the microcontroller. Since the address is read as a group of digital numbers, the switch is connected also to a resistor, whose terminals are connected to V_{CC} and the switch. The DIP second terminal is connected to the ground and the slide state of the switch is what determinates if the sampled state is '0' or '1'. It is possible to sample 64 different values, which means that we can address 64 different and independent zones.

Potentiometer - The most common concept among the lighting system is to have a switch, turning the lamps on/off, but since this is a LED based solution we can actually dim it in an efficient way. Considering this characteristic, it was mandatory to embed a potentiometer in the system. The functional process is quite easy, since the potentiometer is a variable resistor, ranging the voltage value on the resistor terminals from 0 to Vcc, it was only necessary to include the sampled value into the data package.

Validate Key - In this implementation, it was required an initiation button, whose objective is to induce the system into a state, where a new setting can be chosen or modified. When the user is perfectly satisfied with the entered setting mode, he will press the validate key in order to transmit the desirable light profile. After this operation, the system will be awaiting indefinitely for new orders.

Communication Driver - Although this block was drawn being directly connected to the slave board one, it can both represent a wireless and/or wired connection/driver. Bear in mind that this project was designed to support both solutions where they can be perfectively working in parallel. The distinction between addressing a wired and a wireless address will be dictated by the most significant bit of the mode block.

Mode - It was used the same strategy implemented to select the desirable address, on address block. A DIP switch was used to select one of the four modes made available in this lighting system. The most significant bit was used to identify weather the system should transmit the package through the wired or wireless transceiver.

3.1.2. Slave

The slave board, represented on the right side of the figure 17, is the 'heart' of this project. Its role, in the system, is to follow the profiles and settings sent by the master board. After that it works by itself. Making use of the assembled sensors it will be able to sample light values, detect motion and prevent the LED luminaries from overheat. Since it was decided to give the slave boards the maximum independency, they are featured to be used without a master. A possible application scenario is represented on the figure 18. This feature brings this implementation closer to the industry standard in light management systems. Slave board is constituted by six different blocks, which will be described in depth, in order to give reader a global functional idea of the project. (LED driver, temperature sensor, PIR sensor and light sensor are going to be explained in separate sub-section for organizational purposes.)

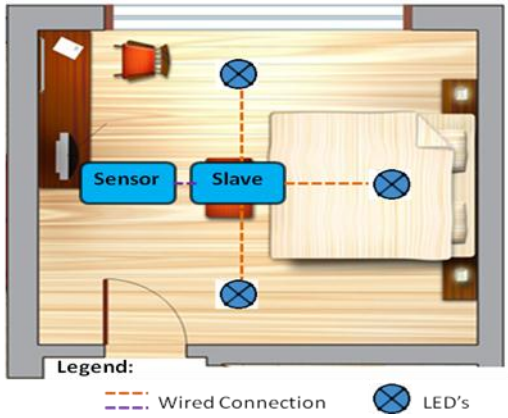


Figure 18: Standalone Implementation

Address block - This block works in the same way as the master block. The only difference is that in the slave board is solely going to read its address in the starting moment of every working session.

Communication Driver - Although this block was drawn being directly connected to a master board, it only represents one of the possible connections: wireless or wired connection. Since a slave board was intended to operate in specific conditions, determined by the environment where it is installed, it only makes sense to be equipped with one of the two technologies/drivers.

3.1.2.1. LED Driver

A LED driver is an electrical device that regulates the power to a LED or matrix of LEDs. The main difference between conventional power supplies and LED drivers is the possibility to adapt to every changing needs of the luminary. As it is known, the increasing of the temperature changes the electrical properties of the LEDs. The increase of temperature increases the current through the LEDs and the increase of current increases the temperature becoming unstable. At the same time, the LED lifetime is directly related to the temperature that it operates. So, regarding the safety of the luminary, it is important to monitor the temperature of the LEDs.

For this project purpose, the LED driver is included in the whole control system, whose objective it is also to provide the correct light level. It will be included, in the sub-section 3.1.2.2., a short description of the temperature sensor used to protect the LEDs.

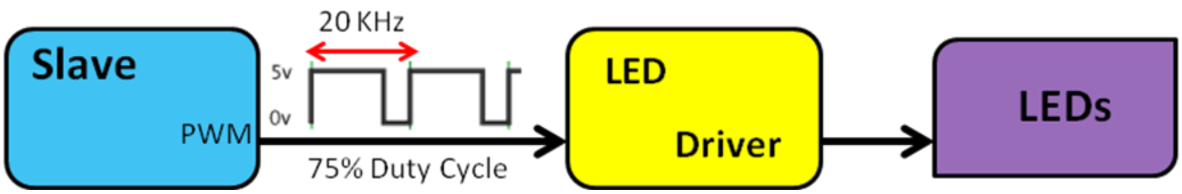


Figure 19:Driving LED Luminary Using a PWM Signal

To drive and control this LEDs implementation was used a constant current power supply based on Texas Instruments LM3424. The LM3424 is a constant current N-channel controller with some relevant features: input voltage from 4.5 to 75 V; PWM and analog dimming; operating junction temperature from -40 ºC to +125 ºC; a thermal foldback for temperature management of the LEDs [24]. Arduino's PWM block was initialized to work at a frequency of 20 kHz (fig.19), in order to meet and respect the LED driver characteristics.

3.1.2.2. Temperature Sensor

A standard procedure in the LED lighting industry is to include overheat protection control on LED luminaries. It is important to state that the LEDs lifetime will be significantly reduced if they operate above the normal heat level.

The LM335 is an integrated temperature sensor based on Brokaw's cell which generates a voltage proportional to the sensed temperature. The conversion ratio is about 10 mV/ºK and it operates from -40 ºC to 100 ºC or approximately from 233 ºK to 373 ºK [25].

According to the LM335 datasheet, the sensor operates correctly when polarized with a current from 400 µA to 5 mA. In this project was used a resistor as a current source (fig. 23), so to correctly dimension it, it was determined the maximum and minimal resistor which would still make the sensor work properly. Since the conversion ratio is 10 mV/ºK, the output voltage can be a value between 2.33 V and 3.73 V. This circuit was supplied with 5 V (Vcc) and using the following expressions it was possible to determinate the resistor range of operation.

$$R_{max} = \frac{Vcc - Vout_{max}}{Imin} = \frac{5 - 3.73}{400 \times 10^{-6}} = 3175\ \Omega \qquad (1)$$

$$R_{min} = \frac{Vcc - Vout_{min}}{Imax} = \frac{5 - 2.33}{5 \times 10^{-3}} = 534\ \Omega \qquad (2)$$

Respecting the impositions shown above, it was chosen a 1 kΩ resistor to polarize the sensor.

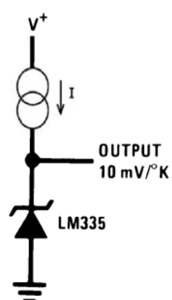


Figure 20: LM335 Electrical Diagram [25]

3.1.2.3. Light Sensors

As a light sensor, it was used a high accuracy ambient light sensor by OSRAM, the SFH-5711[26]. The reason why it was chosen, among the others available in the market, it is because its response perfectly matches the human eye sensitivity and also due to its high accuracy over wide illumination ranges. This light sensor has a logarithmic current output.

A logarithmic response is more suitable to detect small changes in low brightness levels, just where is necessary to act. When measuring high brightness levels only relatively large variations are of interest.

The linear output detectors like phototransistors or photodiodes have changes of the output current which are proportional to changes of the illuminance. The SFH-5711 consists of a photodiode and an IC with the following functions: amplification of the photodiode output signal, logarithmic converter and temperature correction. The final output is according to the expression:

$$I_{out} = S \times \log\left(\frac{Ev}{Eo}\right) \qquad (3)$$

with: S (sensivity) = 10 µA/dec; Ev (lux) – the ambient light illuminance; Eo = 1 lux (value of reference).

In the figure 21, we have the electric diagram of the sensor wired to the microcontroller. In order to develop the necessary algorithm to obtain the actual light level, it is only necessary to reorganize equation (3), sample the correct V_{out} value and compute the equation (4). Bear in mind that the V_{out} value has to be remapped, since the sensor is powered at a 3.3V and the microcontroller at 5 V. Precision will be lost in the process although the measures proofed to have an inferior error than 5%. The readings were compared with Extech HD450 Light Meter measures.

$$E_V = 10^{\frac{V_{out}}{R_L \times S}} \qquad (4)$$

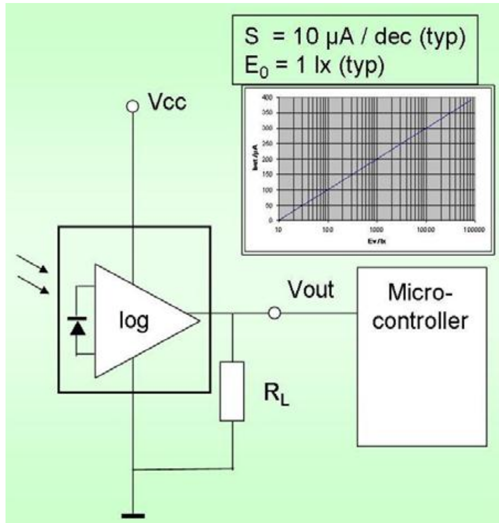


Figure 21: Optical Sensor - Electric Diagram [26]

As can be seen on the graph represented in figure 22, to select a work range of 20 klux, which is enough high to measure all light in industrial scenarios, lead us to a value of the resistor RL = 68 KΩ.

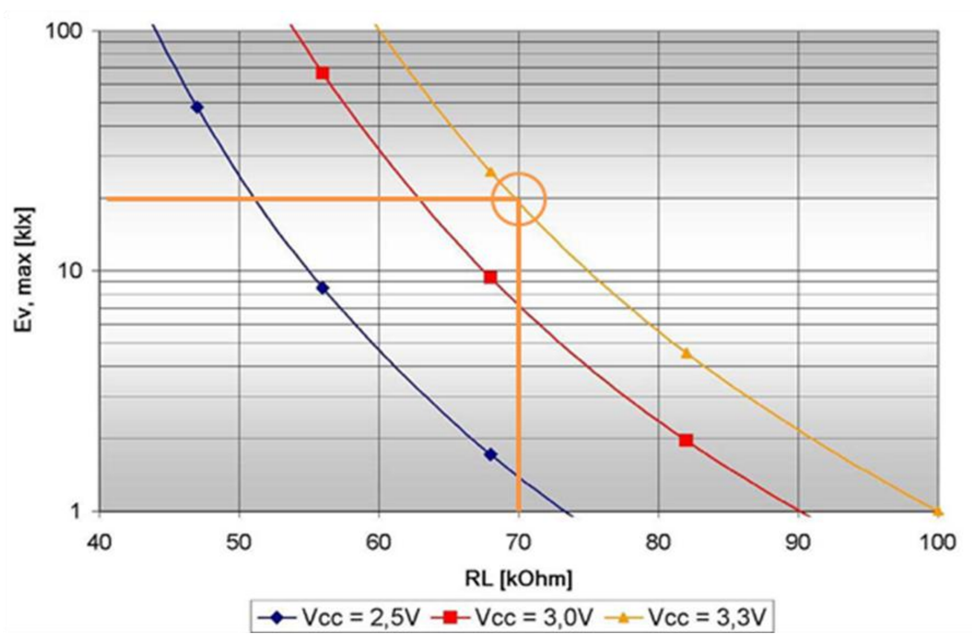


Figure 22: Maximum Detectable Light Level vs. Load Resistance [26]

3.1.2.4. PIR Sensors

In order to detect motion in the monitored perimeter, a PIR Parallax 555 (fig. 23) was chosen. It is an all-built in digital sensor whose functional work is basic. When some motion is detected, sensor output raises from the logical value '0' to '1'.

Global features [27]:

- Wider supply voltage, from 3 to 6 VDC
- Detect a person up to 10 meter away
- Onboard LEDs light up the lens for fast visual feedback when movement is detected

The traditional implementation of PIR sensor in microcontrollers is directly connecting it to its external interruptions pin. Whenever motion is detected by the sensor, it will induce the running program to stop and service the interruption routine.

Figure 23: PIR Parallax 555 [27]

3.2. Arduino

The chosen platform for implementing the designed system was Arduino.

Arduino (fig. 24) is an open-source microcontroller with an Atmel processor, pins for digital input/output and pins for analog input [26]. The concept behind Arduino is to create a development platform simple enough to be used by non-engineers or people whose expertise is not electronics. The most attractive point about Arduino is the huge community who daily shares its projects on Arduino's website. With such an altruistic spirit this community shares their codes and implementations, in order to create one of the biggest code libraries on the internet. Currently, and as an example, Arduino's platform already supports the popular communication protocols such as: CANOpen, X10, DMX512, ZigBee, etc..

Figure 24: Arduino's Platform [28]

The pins of the microcontroller are available with headers to make its use easier. Currently it is possible to purchase shields and add-ons to this platform to control Android cellphones, servo-motors, Ethernet networks, and many more.

Although Arduino can seem to be a complex platform, in its true essence it is nothing more than a microcontroller (ATmega328P [29]) whose pins where made available and programming was made simpler.

Developing and implementing every feature on this project was made very simpler using Arduino. Although Arduino Uno's features (tab. 9) are not the more modern made available today, it is still useful for nowadays application.

Main Features	
Microcontroller	ATmega328P
Operating Voltage	5V
Digital I/O Pins	14
PWM	6
Analog Input Pins	6
Flash Memory	32 kB
SRAM	2 kB
EEPROM	1 kB
Clock Speed	16 MHz

Table 9: Arduino’s Main Features [29]

3.3. Communication

In the previous subsections, it was specified the general design concept an analysed all the master and slave boards constituents. Picking the most suitable protocols was not an easy task, since the goals of this project forced an implementation using a protocol which requires to pay membership fee or to be an authorized member. So the choices were narrowed down to: CANopen, DMX512 and X10, for wired implementation and Hope's protocol/transceiver for wireless implementation. As stated before, PLC based solution are not the most appropriate choices when it comes to work in inductive environments such the industrial ones. Both CANopen and DMX512 are becoming outdated but, since DMX512 was the only one used in lighting systems, it seemed to be the most suitable option. Since the aim of this implementation was to be as versatile and less expensive as possible, respecting all this project goals and restrictions, it was decided to go for the wired solution based on the EIA-485, for the physical media, and, over it, running the DMX512 protocol. In this project it was also decided to use a HOPE's radio frequency transceiver – the RFM12B. It is: a single chip designed to be used in low power applications; a multichannel FSK transceiver designed to be used in unlicensed 433, 868 and 915 MHz bands. This transceiver is a flexible and low cost alternative to a ZigBee solution [18] which is licensed protocols. The RF implementation will be somewhat inspired by the DMX512 protocol, since what is sent in the packages is the PWM value which corresponds to a certain light index. Both wired and wireless implementations virtually work in the same way, only the medium is different. The only relevant difference between both versions is that the RF version has an acknowledged mechanism unlike DMX512 where the communication is unilateral, therefore impossible to support such feature. With this add-on, the project becomes more reliable and robust.

In this implementation it is only possible to address 64 channels/factory zones, although it does not make use of technology full potential it still seems like a reasonable number of different addressable zones.

To clarify what it is sent between devices in the DMX512/RF package is a value in the range from 0 to 255, which means we will have 256 different light levels but we have to realize that these levels have no constant value, since they will change according to the natural light present on the zone. So it is only reasonable to sample the present light value at the time the command was sent and reproduce that very same value in case the system restarts by any reason.

3.3.1. Wired Implementation

The wired implementation is based on the RS-485 transceiver where DMX512 protocol runs over a twisted pair cable. DMX512 protocol is a very simple protocol which allows the control of 512 dimmers in a daisy chain arrangement. The basic signal is a 250 kbps serial signal, eight data bits, one start bit, two stop bits, and a frame start. When DMX512 was created it was not supposed to be used in lighting management systems, so little improvements had to be made. Those improvements can not violate DMX512 principles such as: speed rate or number of data bits have to remain untouchable.

First, let us take a closer look on the DMX512 frame. As it can be seen in the figure 25, all the 512 possible addressable channels receive their correspondent light level in the same frame. Generally speaking, every node will be in "listening" mode, awaiting the incoming light brightness commands, selecting the one intended for its channel. Before developing the overall improvements made on the protocol it is important to bear in mind that only 64 zones are going to be addressed.

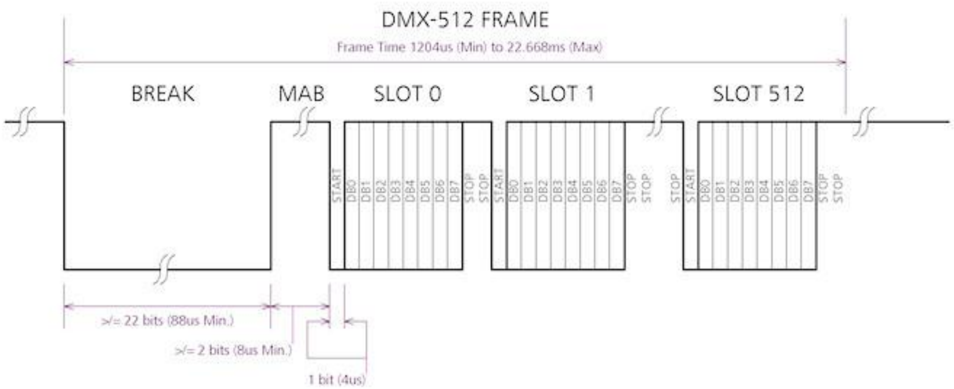


Figure 25: DMX512 Data Frame

The idea behind this protocol hacking is to still transmit the brightness level in the first 64 channels frame leaving unused the other 448 channels. Since only 64 channels can be addressed we can give other meaning to those channels left unused. In this implementation those channels were used by the master to communicate the desirable mode (consult section 4.1, to read more about the implemented functional modes) to a specific factory zone/slave board. The information of how bright a zone must be was not enough to implement a self-sufficient lighting management system. So, at this point, we are only focusing on the 128 first channels.

PWM1 1	PWM2 2	PWM3 3	MODE1 64+1	MODE2 64+2	MODE3 64+3
PWM4 4	PWM5 5	PWM6 6	MODE4 64+4	MODE5 64+5	MODE6 64+6
PWM7 7	(....)	PWM64 64	MODE7 64+7	(....)	MODE64 64+64

Figure 26: Modification Performed on the Transmitted Data

After those 64 "brightness level" channels will come 64 more "channels", from 65 to 128, containing the desirable operating modes. Bear in mind that every single node, to operate correctly, has to pick the information from its designated channel, plus the mode in which it will work. To clarify the modifications performed, it was created an example (fig. 26) of how the transmitted frame looks like. This small modification will allow this system to operate like any other solution on the market.

3.3.2. Wireless Implementation

The wired communication process was explained previously so, now it is the time to implement the wireless version. The concept behind this implementation is to replicate the DMX512 behaviour.

Key features of DMX512 protocol: 512 addressable zones (just 64 required for this project); at least, 8-bit package per channel; 250 kbps speed rate.

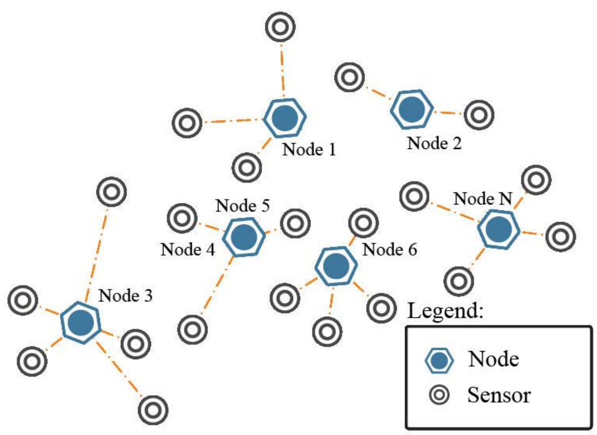


Figure 27: RFM12B Working Diagram

Hardware

The RFM12B is a fully integrated chip with no need of additional hardware. It already includes its own frequency crystal, so it is basically a plug & play implementation.

Network

Hope's transceiver is an oriented network solution; its concept is similar to ZigBee, since nodes can communicate to each other, forming a sensor network. Unfortunately the broadcasting node to node retransmission is not as easy to integrate as in the ZigBee modules. In fact, it is only possible with complex algorithms.

RFM12B network can be composed from 1 to 212 independent sub-networks, which can contain 30 different nodes. It is important to bear in mind two things: the ID of the node should be unique; nodes can only communicate with nodes within the same network in order to avoid losing information.

These features are more than enough to modulate the standalone and distributed system described earlier. Although, the typical application is to use it as a mesh network, to match the wired solution we are going to set the master as node number 1 of every independent network. The slave will always be on node number 2. With this configuration, within each network, we can have up to 28 different sensors attached to the RF network. It is important to state that the master board has to be reconfigured every time we intend to address a new command to a different network. With this implementation, we replicate the unilateral communication system, featured in the wired solution. Unfortunately in this projects it was not explored this technology full potential so, in future projects, it would be appropriate to turn every sensor (motion, lighting and presence) into a different and independent node.

To clarify the model process two figures were drawn (fig. 27 and fig. 28). As it can be seen in the figure 27, the whole system runs independently, having no interaction between nodes. In the modeled system, the master node can interact with each individual zone, working as the interface described in the Distributed System sub-chapter.

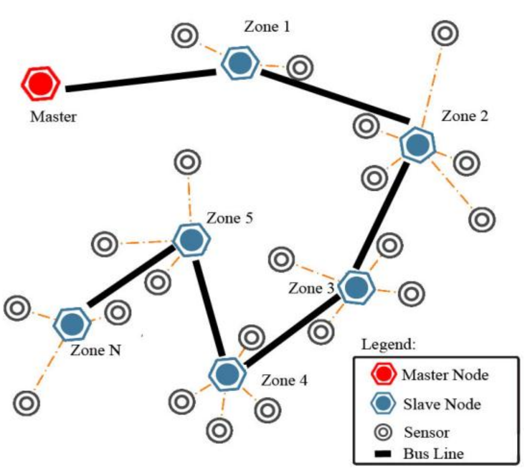


Figure 28: Modifications Implemented on the RFM12B Working Structure

Transmission and Acknowledgment Mechanism

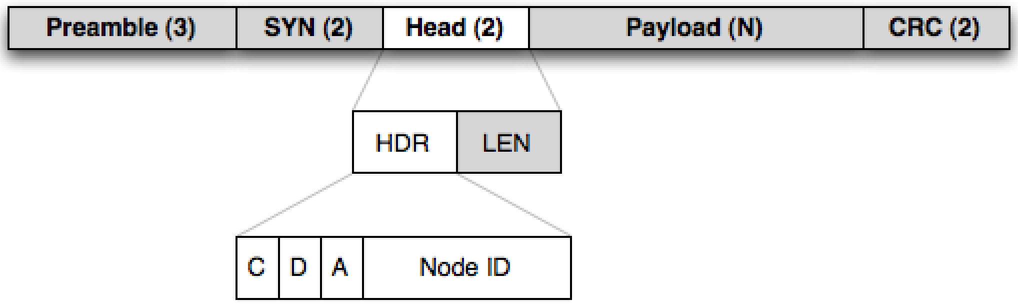


Figure 29: RFM12B Frame Structure [30]

In order to give a deeper insight on the RFM12B communication process, it was included its frame structure (fig. 29). For this project purpose, the most important frame field is the head. These 2 bytes are responsible to identify the node from which the package was sent or, in some cases, from whom was sent, when operating in the broadcast mode. It also discriminates, if the package sent needs or not acknowledgment from the source [30].

HDR (HEADER)

C (CTL) – it is used to send ACKs, and in turn must be combined with the ACK bit set to zero.

D (DST) - Indicates whether the node ID specifies the destination node or the source node. When broadcasting DST = 0, the Node ID field is the source node. For DST = 1, the package is meant to be delivered to a specific node, specified in the Node ID field.

A (ACK) - Whether the package sent needs or not acknowledgment. ACK = 1, requires acknowledgment from the receiver node.

Node ID - When combined with the DST bit, it specifies if it is the destination or source node.

To summarize, the following combinations are used (tab. 10):

Communication Options			
HDR			Action
CTL	DST	ACK	
0	0	0	Acknowledgment not requested
0	1	1	Acknowledgment requested
1	X	0	Acknowledgment packet sent
1	1	X	Combination not used

Table 10: RFM12B Communication Options [30]

The reply packet can also be used as a data packet. This feature possibility may motivate future improvements for this project.

CHAPTER 4:
LIGHTING CONTROL SYSTEM - DESIGN AND TEST

During the third chapter was: explained the global implementation philosophy along with the little details behind the protocols used to turn this project a reality; exposed the potential and limitations of the chosen design and development platform.

From the last chapter remain unexplained which modes/lighting profiles were implemented and how will they influence the selected slave behaviour.

To enrich this dissertation it was included in this chapter content: a SWOT Analysis which compares the existing solutions with the designed implementation, through an analysis of its strong and weak points and also evaluates the threats and opportunities which it will face; the overall cost and a system performance test where the implementation is going to have its performance tested under real life conditions.

Before starting the chapter, let us summarize these solution desirable goals: adjust the light level accordantly; allocate different lighting profiles to different zones; make possible to override system definitions; versatile enough to overcome different physical dispositions.

4.1. Functional Modes

This project backbone is the different lighting profiles which a factory zone can have. Through the investigation done previously, to start conceiving the final design, explained and written on the state of the art chapter, we have some standard lighting profiles such as: daylight level control; occupancy-based control or scheduled control. From those stated before were picked the most relevant ones, those which a lighting management system cannot live without or which are more suitable to work efficiently in an industrial environment.

To make the functional modes explanation easier, it was created a table (tab. 11) which gather all the implemented lighting profiles and two statecharts which represent the master and slave perceptive when selecting/selected a certain profile. To contextualize, statecharts are a way to structure an application into logic states. When designing an application it is important to think about: all the possible states; how the application can transition from one state to another; the root state, the initial state of the application; the way between the initial state and exit state. Meanwhile, as the problem is better known, some redundant states will be possible eliminate. The exercise on thinking about these issues will make easier to plan and develop the project [31].

As it can be seen by analysing table 11, in the reality this system only has 4 different modes, being the last bit a way to differentiate the wired addresses from the wireless ones. This will allow no mistakes between both implementations maintaining the system running with no problems or disruption.

Lighting Profiles			
Bit 2	**Bit 1**	**Bit 0**	**Mode Name**
0	0	0	Wired address - Override
0	0	1	Wired address - New Threshold
0	1	0	Wired address - Standalone
0	1	1	Wired address - Shutdown
1	0	0	Wireless address - Override
1	0	1	Wireless address - New Threshold
1	1	0	Wireless address - Standalone
1	1	1	Wireless address - Shutdown

Table 11: Master Board Functional Modes

In order to give the reader a deeper understanding on how whole the system works, were created two sub-sections where the master and slave perspectives were exposed.

4.1.1. Functional Modes Description

Override Mode - In every lighting management system it is indispensable to have an override to the current setting. It is not due to the lack of confidence in the rest of the modes but it is necessary to always have a way to force the system operating at a determinate light level. Even when using appropriate studies, sometimes determinate tasks require appropriate light, in order to make the worker confident/comfortable enough to do them.

Standalone Mode - Inspired by the study performed at the state of the art chapter, the standalone mode was implemented in this solutions, due to be the industry standard in nowadays lighting systems. This lighting profile will be able to harvest natural light, if possible, in order to meet a determinate light level. Along with this policy, whenever presence is not detected within the monitored area, these sector lamps will be turned off. This way, the electricity saving potential of this project will be unleashed, since the system will automatically save energy when no one is in the zone; whenever someone is detected, the energy wasted is going to be minimal, because it will only spend enough to readjust the light level until it meets the threshold. The desirable threshold is set by user using the new threshold mode and this means that the only data package sent is the chosen lighting profile. This procedure will allow the slave to keep a desirable light threshold saved in its memory.

New Threshold Mode - This lighting profile sets the new threshold which the standalone mode must meet. The slave modules come predefined from the factory to work in standalone mode at 400 lux. This value was chosen to be a light level suitable for the execution of general tasks in industrial environment or in an office, a call-center and, in general, in environments where an average visual acuity is required [32]. If the user decides to change this preset value it has to do it through the new threshold mode. No other action is performed in this mode.

Shutdown Mode - The name of this mode self explains its behaviour which is turning off the luminaries in the selected zone. At first, this mode can seem a little redundant, since we can achieve the same result using the override mode but, featuring this propriety, was getting it closer to the contemporary lighting systems.

4.1.2. Master Perspective

After defining the functional modes, it is time to describe them under the master board perspective. For an easier perception, it was created a statechart diagram, shown in the figure 30. Taking a closer look on the state diagram we can see two main paths: the override/ new threshold and the standalone/shut down. Before stating what is different between them, it is important to realize what they have in common.

Whenever the master board starts up it will be in the 'IDLE' mode and it will remain in such state until the 'validate key' (chapter 3.1.1) is pressed. Before pressing the key, user must select, on the DIP switches, which mode and address wants to select.

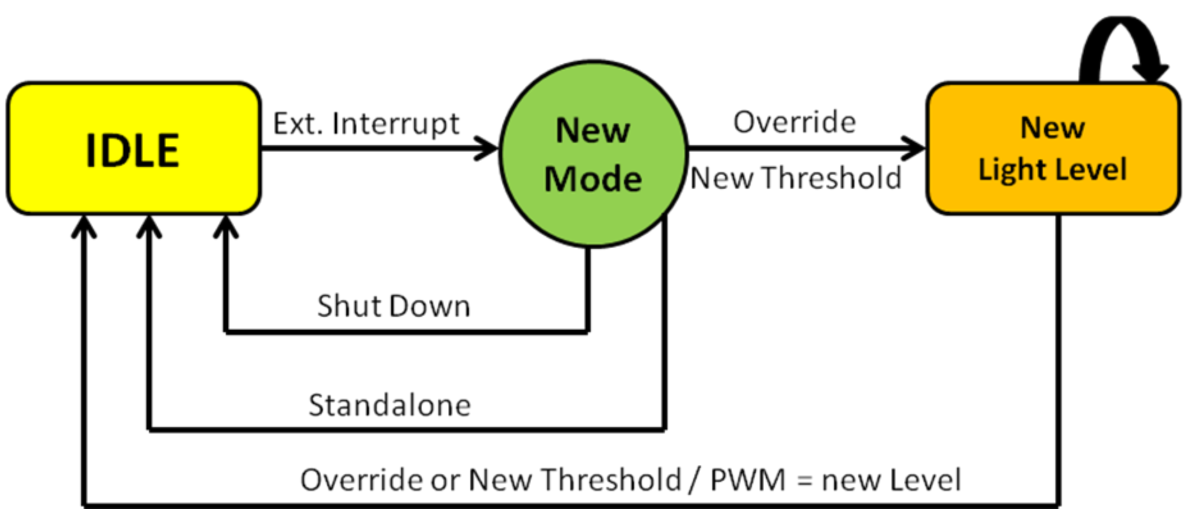


Figure 30: Master Board StateChart

After pressing 'validate key', the system will be interrupted and entering the 'New Mode' state. At this point, four possible routes can be taken, although two of them are virtually the same.

Override and New Threshold Path

After one of these two modes has been selected, master board will add the mode 'code' into the selected slave board address spot + 64. It is important to clarify that brightness levels will be added into the selected address spot and the light profiles into the selected address spot + 64. In the end of this procedure, the system will return to the 'IDLE' state.

Example: To illustrate the changes performed on the DMX512 frame and to enrich this explanation it was created a representation (fig. 31) of DMX512 frame when addressing to the third zone the 'override' mode with '100' of brightness. It was not defined which media was chosen, since it was irrelevant for example.

Standalone and Shutdown Path

After one of these two modes has been selected, master board will add the mode 'code' into the selected slave board address spot + 64. There is no need to add any brightness level in the selected channel spot, since the brightness level on the shutdown mode will be 0 and on the standalone will be defined by the new threshold or it will use the factory default value, saved on the slave board. In the end of this procedure, the system will return to the 'IDLE' state.

Example: To illustrate the changes performed on the DMX512 frame and to enrich this explanation it was created a representation (fig. 31) of DMX512 frame when addressing to the forth zone the 'standalone' mode. It was not defined which media was chosen, since it was irrelevant for example.

PWM1 1	PWM2 2	100 3	MODE1 64+1	MODE2 64+2	00 64+3
XXX 4	PWM5 5	PWM6 6	10 64+4	MODE5 64+5	MODE6 64+6
PWM7 7	(....)	PWM64 64	MODE7 64+7	(....)	MODE64 64+64

Figure 31: DMX512 Modification Example

4.1.3. Slave Perspective

As in the master perspective sub-section, it was created a statechart diagram (fig.32) with the slave perspective of override and standalone modes, since they were the more complex ones. Before stating what is different between them, it is important to realize what all modes have in common.

Whenever the slave board starts up, before going to the 'IDLE' state, it will check for previous saved operating definitions. In case it is the first time turned on, it will go straight 'IDLE' state awaiting for incoming DMX512/RF frames. In case the frame does not contain any data intended for the slave it will remain in the 'IDLE' mode. The same principle is applied when the data/mode received is the same as the frame received before.

Shutdown

Although the shutdown mode is not represented in the figure 32, when it is selected, the slave board turns the allocated lamps off. In the end of this procedure, the system will return to the 'IDLE' state.

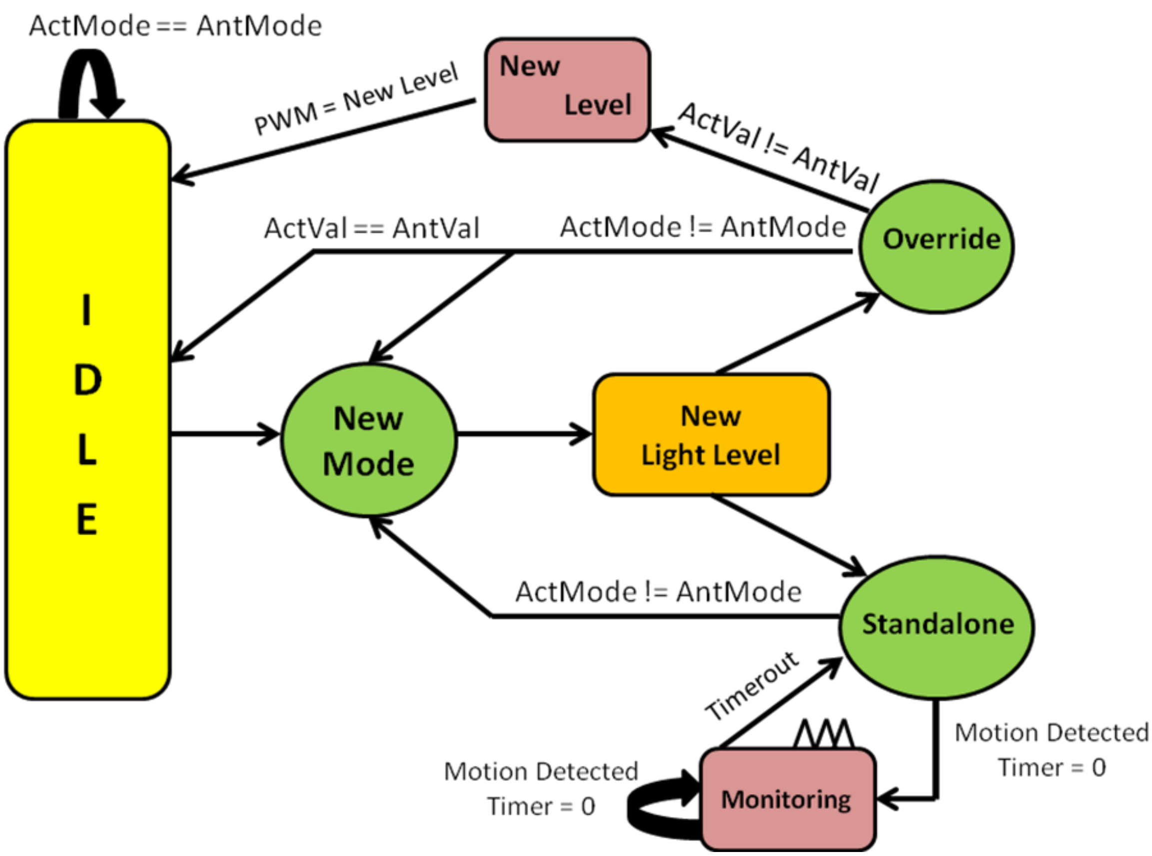


Figure 32: Slave Board StateChart

Override and New Threshold Path

The override and new threshold modes operate virtually in the same way, since the idea is to set a new light level in that zone, regardless of the mode that it is going to operate. Master board will send as many data fames as needed until the user is pleased with the light level on addressed sector. Since the algorithm is based on loops, until the desirable light level is established, the program will be verifying if the income data has changed and, in case that it changed, the slave will apply it on the luminaries (fig. 32).

Standalone Path

When selecting the standalone mode, the system will remain in a hiatus state, until presence is detected or master order it to change its functional mode. If in the monitored perimeter is detected motion, a timer will start counting; light level will be sampled in order to dim the lamps accordantly to meet the predefined threshold. Every time motion is detected the timer will reset. In case the timer expiries, meaning that motion was not detected for a certain period, the system will return to a hiatus state.

To finalize this slave perspective, it is important to state that, after every new setting change, the slave will save the current mode and light level, if needed, in the EEPROM.

4.2. Overall System Cost

In the objectives of this dissertation was the need for this project to be as inexpensive as possible. The reason behind this objective is to turn the investment return time as shorter as possible and also to decrease the high initial cost of implementing a LED based solution.

Summing these two factors to the need to be as versatile as possible, it is evident that, opting to develop this project with Arduino's platform, was one of the less expensive and elegant solutions available. In this sub-chapter, it is going to be shown the full cost of the project presenting the costs of master and slave board separately. It is important to state that the price of replacing the previous lighting installation was not accounted, due to its high complexity. Such study, by itself, might be ample enough to be an independent thesis project.

To assemble the developed prototype it was used the Arduino Uno board which was intentionally designed to easily support external shields and add-ons. To respect this philosophy, this project prototype was conceived as an external shield, although, if this solution is really meant to be mass produced, a different design has to be created (different from the one exposed on the 4.5 sub-chapter).

This new design would allow this project to save even more, since it would solely use the ATmega328P microcontroller and not the complete Arduino Uno board. Considering this last factor, it was created a table of costs focusing these two approaches. The first one includes the whole Arduino platform and in the second version only the microcontroller was accounted. There is no other difference between both versions.

Master Board				
Part Name	**Part Number**	**Price (€)**	**Nº**	**Source**
MAX487CPA+	1188013	1,44	1	http://pt.farnell.com/
RFM12B	1878282	9,96	1	http://pt.farnell.com/
DIP-Switch	1363071	0,93	2	http://pt.farnell.com/
SPST Switch	2079610	1,17	2	http://pt.farnell.com/
RJ45 Socket	3938359	0,85	2	http://pt.farnell.com/
Miscellaneous	-	5	1	http://pt.farnell.com/
Arduino Uno Rev. 3	1848687	26,05	1	http://pt.farnell.com/
Total Version 1	**€ 48,35**			
ATmega328P	1715487	1,70	1	http://pt.farnell.com/
Total Version 2	**€ 24**			

Table 12: Master Board Assembling Cost

To give the reader a better understanding, were created two separate tables, referring to master (tab. 12) and slave (tab. 13) cost boards. It was not accounted the enclosure boxes expenses, since it was not designed or chosen one in specific.

Slave Board				
Part Name	**Part Number**	**Price (€)**	**Nº**	**Source**
MAX487CPA+	1188013	1,44	1	http://pt.farnell.com/
RFM12B	1878282	9,96	1	http://pt.farnell.com/
DIP-Switch	1363071	0,93	2	http://pt.farnell.com/
SPST Switch	2079610	1,17	2	http://pt.farnell.com/
RJ45 Socket	3938359	0,85	2	http://pt.farnell.com/
OSRAM SFH-5711	1573496RL	1,41	1	http://pt.farnell.com/
Parallax 555	555-28027-ND	8,07	1	http://www.digikey.pt/
Miscellaneous	-	5	1	http://pt.farnell.com/
Arduino Uno Rev. 3	1848687	26,05	1	http://pt.farnell.com/
Total Version 1	**€ 57,83**			
ATmega328P	1715487	1,70	1	http://pt.farnell.com/
Total Version 2	**€ 33,48**			

Table 13: Slave Board Assembling Cost

4.3. SWOT Analysis and Final Statements

SWOT stands for Strengths, Weaknesses, Opportunities and Threats which is a very well-known tool amongst the management field of expertise. The SWOT analysis is a tool based on an evaluation model of the competitive position of a company or product in the market. This competitive evaluation is always a part of a bigger plan such as a marketing or business strategy. The SWOT analysis is divided in two factors: the internal factors, which are the ones that can be controlled by the company or the user and external ones which cannot be controlled such as the government or the competitive market [33].

In a brief approach:

- **Strengths** - are the controllable variables which have a direct impact in the success of a product. It is desirable to have as many positive points as possible.
- **Weaknesses** - are the negative aspects about a product which will play a big role in the overall failure. The goal is to quickly improve these points turning into strengths.
- **Opportunities** - are the environmental conditions that can be positive for the success of the designed product.
- **Threats** - are the environmental conditions that can block the success. Bear in mind that every opportunity can turn into a threat and vice versa if not predicted. The difference between them is not always so evident.

In the end of this analysis, it is supposed to have a better understanding of how to compete successfully in a market niche.

Nowadays, the most popular lighting solution, used in industrial environments, is the metal halide lamps. In order to implement such a system like the one designed in this dissertation, the company has to invest in a new overall lighting system based on LED lamps. Although the developed lighting management system and implementation costs are fairly low, the demand to replace the previous technology can represent a huge barrier in the project success. Unfortunately, we are facing a financial crisis period which can block the proliferation of newer technologies. Companies are forced to cut on their budgets and this may represent other obstacle for this project success. The environmental policies taken in the last few years are highly favorable to the development of this kind of products, since the green technologies are turning to be more and more attractive. One of the biggest threats that this implementation faces is the market competition. Companies like Legrand and OSRAM have already launched similar products to the market, so where can this implementation beat the competition? This project was built with the purpose to be the less expensive solution in the market, granting equal performance as the competitors. The use of both wired and wireless connections is certainly a plus, since no other offers this features on this price range. Changing to LED lamps has such a high cost that can constrain the overall interest in this kind of system. Unfortunately with the protocol chosen for the project is not possible to track the overall consumption and performance. Nowadays, many domotic applications are making use of Ethernet connectivity to monitor and control the implemented systems. This kind of features makes them more suitable to be used in modern/high-end buildings. In the figure 33, is shown the SWOT analysis which summarizes, in a graphic way, the pros and cons of the designed lighting management system.

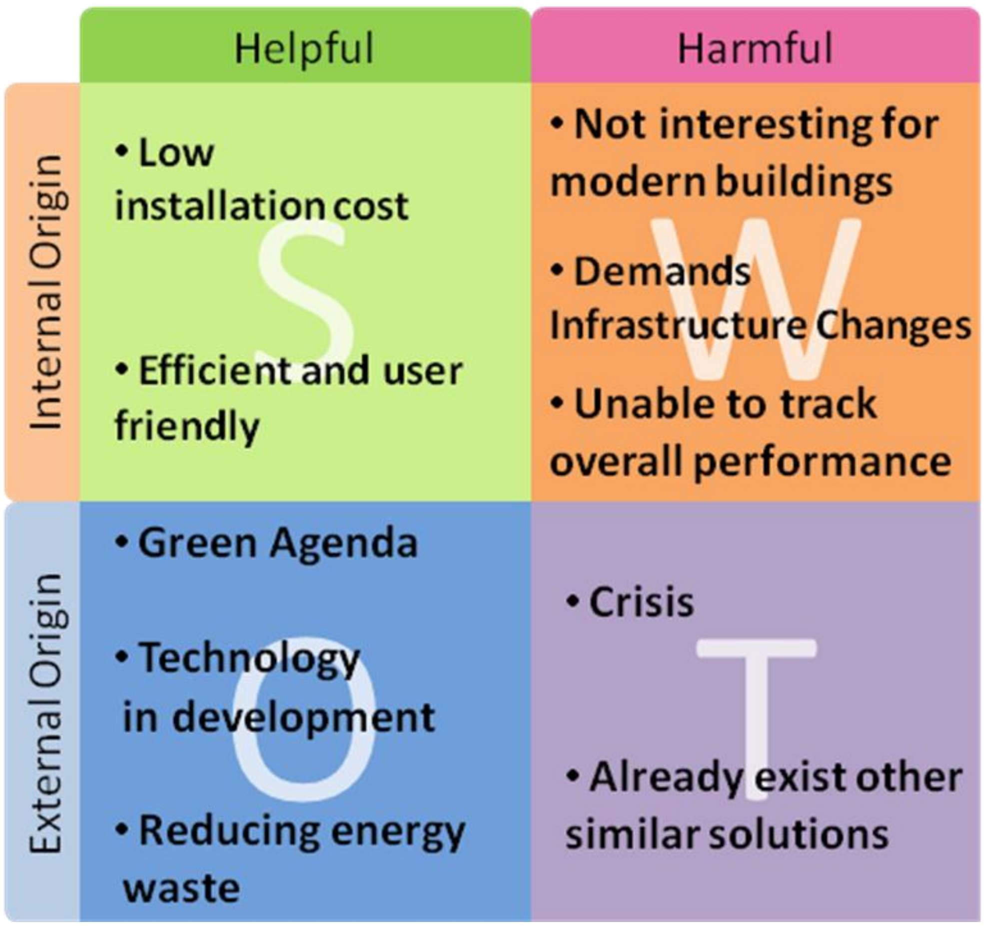


Figure 33: SWOT Analysis

4.4. System Performance Test

In the very last phase of this project it was devised an experiment to put this system under test. The main objective was to observe this project overall behaviour and performance under a real life setting.

In order to simulate a real factory compartment, it was designed a small enclosed box directly facing ambient light with just one of its faces. The prototype and light sensor were placed inside the scale model, being influenced with nothing but natural and artificial light. To collect the sampled data, a computer was connected to the prototype serial output port, which periodically sends values. These values were stored in a spreadsheet enabling posterior treatment and analysis. Although the recreated setup was far from being a professional implementation, it allowed to gather the necessary data and also allowed to conclude that the light sensor used has an error inferior to 5% of the nominal displayed value on the professional Extech HD450 Light Meter. In order to accomplish this experiment it was also needed a power supply to power the LED driver, as it can be seen on the left side of figure 34. In the right side of figure 34, it is possible to see the system running, from the outer side of the windowpane.

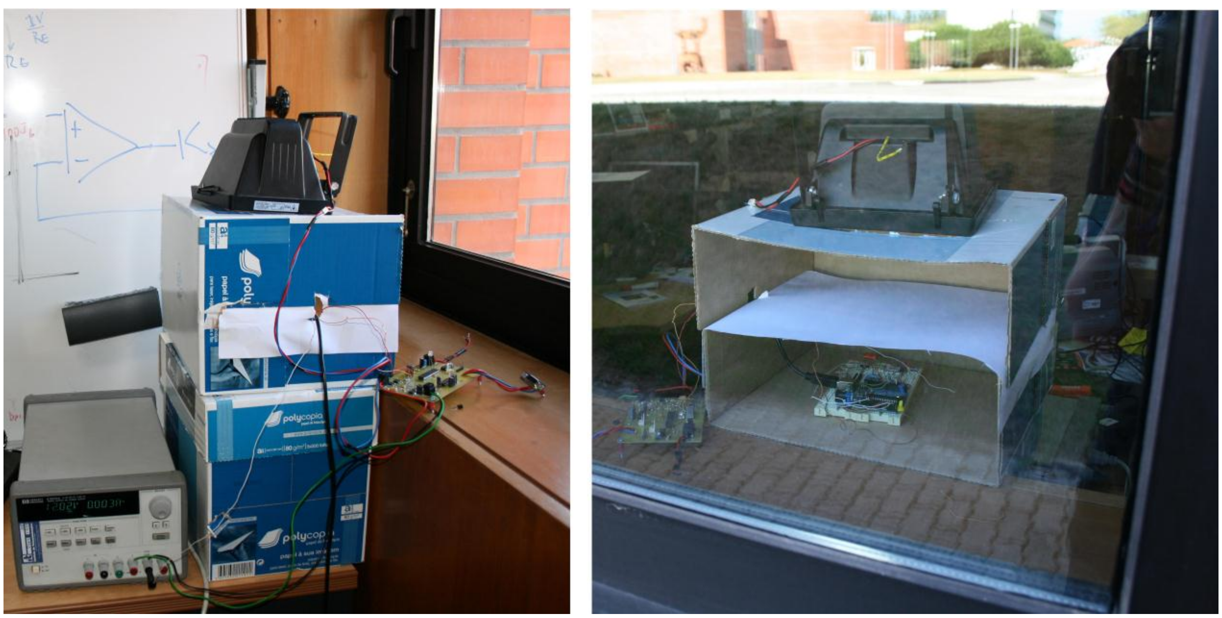


Figure 34: Experiment Set-up

The experiment was realized during a bright September day, from 15:00 to approximately 04:00. This schedule was chosen in order to test the system in a sunset setting. The sapling rate of the test was one sample per minute and during the test it was sampled the light level, in lux, and the PWM value which was directly used to attack the LED driver. The system was set to maintain a constant level of 400 lux ±10%. This value was chosen to be a light level suitable for the execution of general tasks in industrial environment or in an office, a call-center and, in general, in environments where an average visual acuity is required [32].

When the test started at 15:00 the set-up value was lower than the existing light level but high enough to keep the system running even after the sunset. This procedure allowed a full diagnosis of the system behaviour, testing it in every possible stage. As it can be seen in the plotted graph (fig. 35), the light level starts decreasing at 17:00. The system triggers approximately 230 minutes after the start and, after a transitional period, a stationary period was reached around the 400 lux intended value. It is important to state that this test was run without any occupancy/motion sensor, its goal was solely to test its behaviour in a real situation. It is also important to state that the PWM signal is inverted, which means the luminary will be turned off when the PWM value is 255.

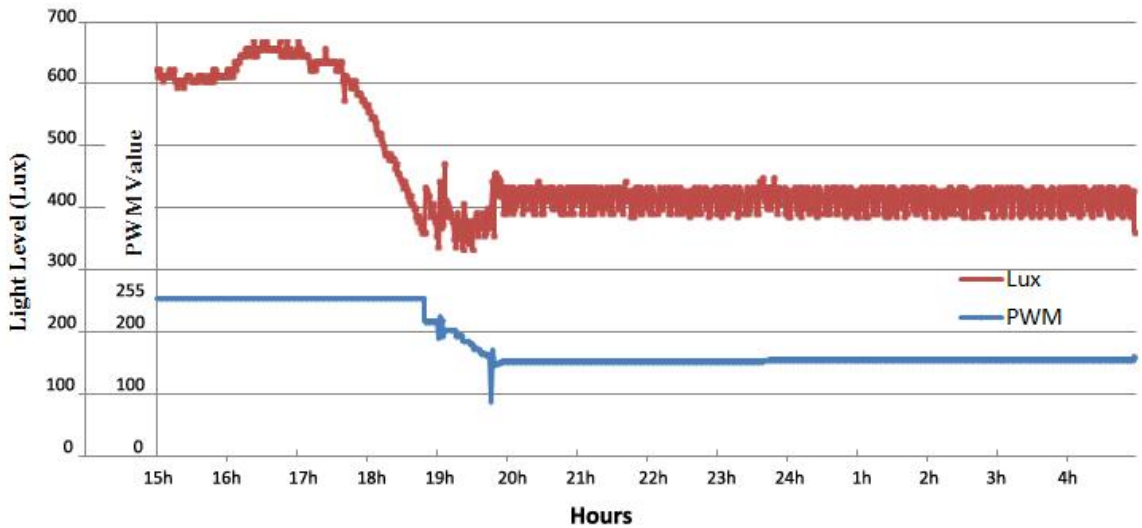


Figure 35: Plotted Data Gathered During the Experiment

CHAPTER 5: CONCLUSIONS

The main goal of this project was to create a lighting management system capable of covering as many scenarios possible at an inexpensive price. This goal was achieved in its total. Since it was desirable the usage of free protocols, the DMX512 seems to be a suitable option. The protocol is starting to become outdated and condemned to disappear, although there is in the market a large quantity of products using it.

The designed solution is capable of operating in all industrial environments, as desirable, and is able to adjust itself according to what is presented. The possibility to turn up a slave board into the standalone mode adds to this system one unique versatility, no other solution on the market has this very same feature.

In order to do a conscious reflection about the designed solutions, it will be presented two versions of possible improvements. One, focusing in what could be done using the assumptions stated in the first chapter. In a second moment, the focus will be put in what would be enriching the project, if only the created design was kept.

Using a microcontroller which featured more ports would enable the possibility to address more zones. There is no real use of such feature but, the marketing impact of advertising 128 or 256 separate zones instead of the existing 64, would definitively be a plus.

Encrypting the RF data packages is a really necessary improvement. The RF implementation is really simplistic at the moment. If this project had the aspiration to turn into a real life solution, the transferred data should be encrypted. For example, imagine a bank scenario, if the robbers would like to shut down all the existing lights they would only need to send a wireless command to override the existing setting.

For the future would be necessary to modify this project and use a more modern protocol such as KNX. The possibility to expand this solution to the intelligent house market is highly attractive.

Focusing in what it seems to be the future for industrial installations it would be improving this project if we added an Ethernet connection to the system to enable the possibility to monitor the overall performance using the internet. If it was created a specific application, it would also be possible to monitor the performance through the new smartphones and tablets.

Since University of Aveiro had already, in the past, developed and accomplished so many projects in the field of lighting management systems, one valuable idea would be creating a specific room, within the telecommunications institute building, where LED/HID solutions could be really put into a test.

In this room the future and the already existing prototypes could be properly tested, in order to start gathering real reliable data. Until now, it is only possible to make some assumptions of how much these systems would save, since there is no assembled infrastructure to test them. In it would also be possible to test in depth control algorithms and lighting management strategies for long periods.

Creating this space, would concede a serious character to the academic works developed. This overall performance studies will allow, in the future, the possibility to elaborate real investment return predictions, which would be very useful to attract new companies/partners to collaborate with telecommunications institute projects on this field.

REFERENCES:

[1] Yoshida, M. (2009). Low-Priced Products Shake the Market; New Makers Vie with Old, Intensifying Competition. Nikkei Electronics Asia. Japan.

[2] Executive, H. a. S. (1997). "Lighting at Work." **1**.

[3] International Energy Agency, Light's labours lost: Policies for energy-efficient lighting, in support of the G8 plan of action,[2006, p. 558.

[4] Solid-state lighting research and development portfolio: Multi-year program plan FY'07-FY'12, Navigant Consulting, Inc., 2006, prepared for Lighting Research and Development Building Technologies Program, Office of Energy Efficiency and Renewable Energy, U.S. Dept. of Energy.

[5] Solid-state lighting research and development portfolio: Multi-year program plan FY'07-FY'12, Navigant Consulting, Inc., 2008, prepared for Lighting Research and Development Building Technologies Program, Office of Energy Efficiency and Renewable Energy, U.S. Dept. of Energy

[6] U.S. Lighting Market Characterization, vol. I, National Lighting Inventory and Energy Consumption Estimates. Washington, D.C.: U.S. Department of Energy, 2002, Navigant Consulting, Inc. [Online]. Available: http:// www.eere.energy.gov/buildings/info/documents/pdfs/lmc_vol1_final.pdf

[7] Solid-state lighting research and development: Multi-year program plan U. S. Department of Energy, April 2012, p. 47.

[8] Machnicki, K. M. (2011). Sistema de Gestão de Iluminação em Ambientes Industriais. Departamento de Electrónica, Telecomunicações e Informática. Aveiro, University of Aveiro. Master.

[9] What is Power Line Communication?, EE Times University, August 2011. Available: http://www.eetimes.com/design/industrial-control/4218852/What-is-Power-Line-Communication-?pageNumber=1

[10] Maxim Integrated Products (2012). Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers Datasheet.

[11] Maxim Integrated Products. (2006). How Far and How Fast Can You Go with RS-485.

[12] Directiva 2008/432/CE, de 23/05/2008.

[13] Building Technologies, HVAC Products, Datasheet KNX bus, Siemens, Switzerland, 2008

[14] Weinzierl, Thomas: KNX-RF A new Standard for Wireless Networks within Buildings; Weinzierl Engineering GmbH, July 2005

[15] http://www.dali-ag.org/

[16] http://www.dmx512-online.com/

[17] Flowers, David; Yang, Yifeng: Microchip MiWi^{TM} Wireless Networking protocol Stack; Microchip Technology Inc.; 2010; available on pdf: http://www.newark.com/pdfs/techarticles/microchip/AN1066_MiWiAppNote.pdf

[18] LTD, H. M. C. (2006). "RFM12B Module Datasheet."

[19] Patrick Kinney. Zigbee technology: Wireless control that simply works, 2003

[20] *Collins English Dictionary - Complete & Unabridged 10th Edition*. Retrieved November 07, 2012, from Dictionary.com website: http://dictionary.reference.com/browse/sensor

[21] "sensor". Oxford Dictionaries. April 2010. Oxford Dictionaries. April 2010. Oxford University Press. 07 November 2012 <http://oxforddictionaries.com/definition/english/sensor>.

[22] WattStopper, WS- 10-28315, rev. 04/2010, Santa Clara, CA, USA

[23] Legrand (2011). Design and Installation Guide, Legrand. **1**

[24] Instruments, T. (2009). "LM3424 Constant Current N-Channel Controller with Thermal Foldback for Driving LEDs."

[25] Instruments, T. (2004). LM135/LM235/LM335, LM135A/LM235A/LM335A Precision Temperature Sensors.

[26] OSRAM (2006). "High accuracy Ambient Light Sensor SFH 5711."

[27] Parallax Inc., Ed. (2007). PIR Sensor (#555-28027).

[28] http://www.arduino.cc

[29] Atmel Corporation (2011). "ATmega48A/PA/88A/PA/168A/PA/328/P."

[30] http://jeelabs.org/2011/06/09/rf12-packet-format-and-design/

[31] Harel, D. (1987). Statecharts: A Visual Formalism For Complex Systems.

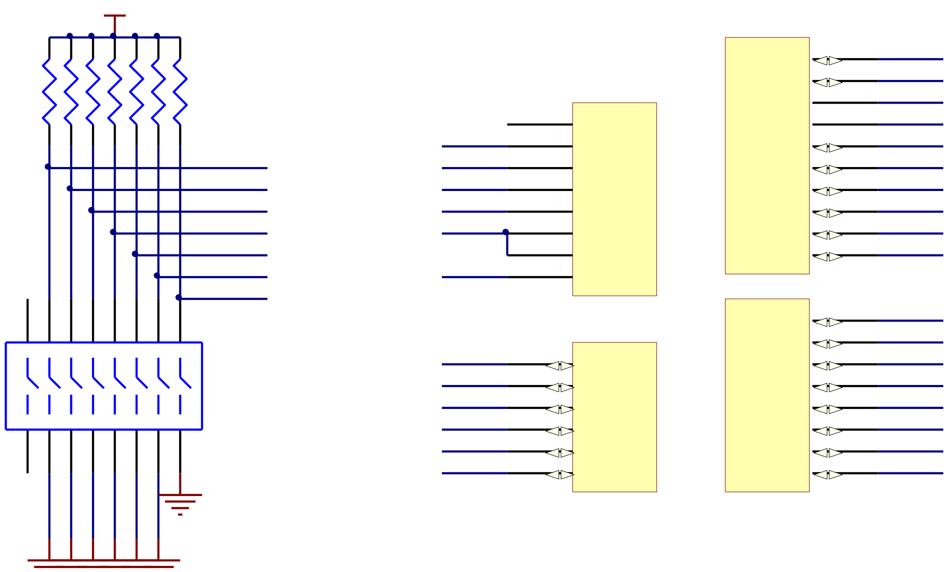
[32] Luminotecnia 2002, Indalux Iluminacion Tecnica, S.L., Valladolid, p. 158-163

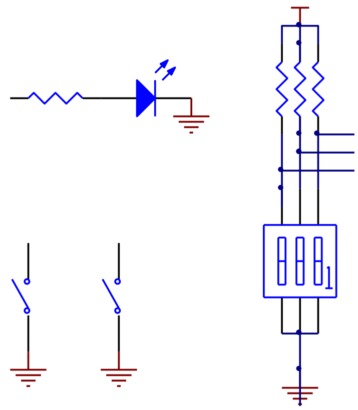
[33] http://www.marketingteacher.com/lesson-store/lesson-swot.html#

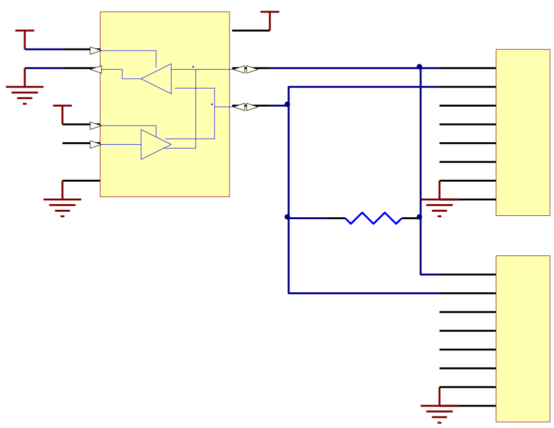
APPENDIX A:
CIRCUIT DESIGN

In order to design the schematic for both boards of this project it was used the trial version of Altium Designer 10.0. Altium Designer is one of the best softwares in the market when it comes to design PCBs and schematics .

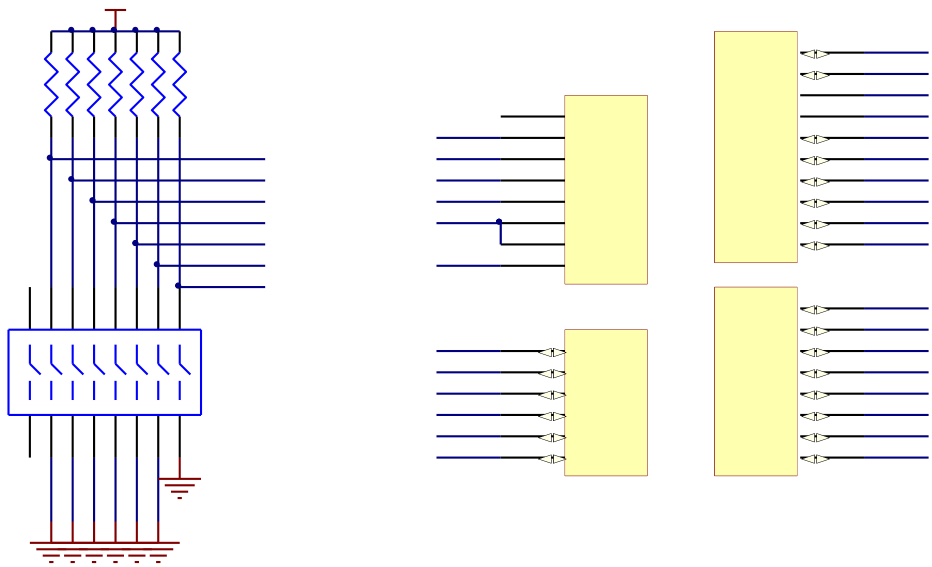
Master Board

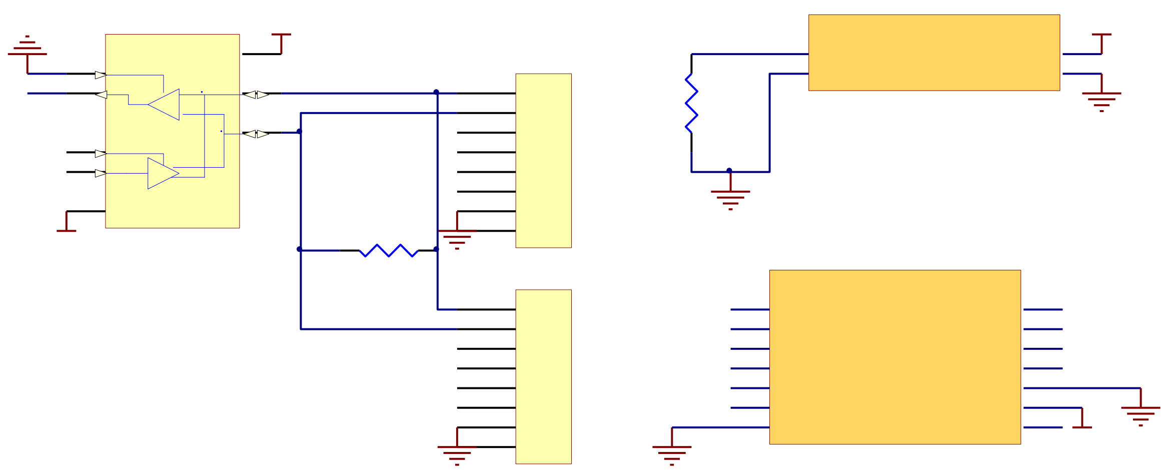






Slaver Board





APPENDIX B:
SOFTWARE DESIGN

The following software was developed using Arduino 1.0.1. This code was implemented and tested on Arduino Rev. 3 boards.

Since this project was developed in two different versions: standalone and distributed. The distributed version will also have two sub-different versions: slave board using wireless and wired connections.

Distributed Master Board:

```
#include <JeeLib.h>
#include <Ports.h>
#include <RF12.h>
#include <RF12sio.h>

#include <PinChangeInt.h>

// Libs used
#include <DmxSimple.h>
#include <EEPROM.h>

#define NO_PORTC_PINCHANGES
#define NO_PORTB_PINCHANGES

//Global Variables
int maxChannels = 256;
int DmxPin = 3;
boolean TRUE = 1;
boolean FALSE = 0;
int overide = B00;
```

```
int newThreshold = B01;
int standAlone = B10;
int RESET = B11;
int led = 4;
int temp=0;
int flag;

MilliTimer timer;

void setup() {
  DmxSimple.usePin(DmxPin);         //config DMX512 with 128 channels on Pin 11
  DmxSimple.maxChannel(maxChannels);
  pinMode(led,OUTPUT);                //Initialize LED status PIN

 rf12_initialize(1, RF12_868MHZ, readChannel());      //Initialize RF Module

  pinMode(8, INPUT);                     //Initialize Int Pin's
  PCintPort::attachInterrupt(8,newSetting , FALLING);
  pinMode(1, INPUT);
}

void loop() {
            //in the limbo
}

void newSetting(){
PCintPort::detachInterrupt(8);
 int mode=readMode();
 int channel=readChannel();
 int lightLevel;

  switch (mode){
    case B000:                       // OVERIDE DMX512
       noInterrupts();
         digitalWrite(led,HIGH);
           lightLevel=newLightLevel512(channel,mode);
          // EEPROM.write(channel,mode);
           DmxSimple.write(channel, lightLevel);
           DmxSimple.write(channel+128,mode);
```

```
    ledBlinking(250);
  interrupts();
break;

case B001:                // new threshold DMX512
  noInterrupts();
    digitalWrite(led,HIGH);
      lightLevel=newLightLevel512(channel, mode);
      EEPROM.write(channel,B010);
      DmxSimple.write(channel, lightLevel);
      DmxSimple.write(channel+128,mode);
    ledBlinking(250);
  interrupts();

break;

case B010:                // StandAlone DMX512
 noInterrupts();
   digitalWrite(led,HIGH);
     EEPROM.write(channel,mode);
     DmxSimple.write(channel+128, mode);
   ledBlinking(2000);
 interrupts();
break;

case B011:               // Shutdown DMX512
 noInterrupts();
   digitalWrite(led,HIGH);
     EEPROM.write(channel,mode);
     DmxSimple.write(channel+128, mode);
   ledBlinking(2000);
 interrupts();
break;

case B100:               // Overide RF12
   digitalWrite(led,HIGH);
     rf12_initialize(1, RF12_868MHZ, channel);
     lightLevel=newLightLevelRF(mode);
     wantACK(lightLevel,mode);
```

```
        EEPROM.write(channel,mode);
      ledBlinking(250);
    break;

  case B101:                  // newThreshold RF12
    digitalWrite(led,HIGH);
      rf12_initialize(1, RF12_868MHZ, channel);
      lightLevel=newLightLevelRF(mode);
      wantACK(lightLevel,mode);
      EEPROM.write(channel,B110);
    ledBlinking(250);
    break;

  case B110:                  // StandAlone RF12
    digitalWrite(led,HIGH);
      rf12_initialize(1, RF12_868MHZ, channel);
      wantACK(0,B110);
      EEPROM.write(channel,mode);
    ledBlinking(500);
    break;

  case B111:                  //Shutdown RF12
    digitalWrite(led,HIGH);
      rf12_initialize(1, RF12_868MHZ, channel);
      wantACK(0,mode);
      EEPROM.write(channel,mode);
    ledBlinking(500);
    break;
 }
    PCintPort::attachInterrupt(8,newSetting, FALLING);
}
```

void wantACK(int level, int mode)

```
{
  interrupts();
  char teste[2];
  teste[0]=level;
  teste[1]=mode;
  for(int i=0; i<10; i++)
```

```
      {
        while (!rf12_canSend())
          rf12_recvDone();

      rf12_sendStart(0x21,teste, (sizeof teste)+1); // ACK Pedido

      timer.set(20);
      while (!timer.poll())
      {
        if (rf12_recvDone() && rf12_crc == 0)
          {
           i=10; // ACK RECEIVED
          break;
          }
      }
      }
}
```

int newLightLevelRF(int mode){

```
  interrupts();
    char teste[2];
    teste[1]=mode;
    flag=1;

    PCintPort::attachInterrupt(1,flagParty, FALLING);

    while(flag==1)
      {
        while (!rf12_canSend())
          rf12_recvDone();
          teste[0]=lightLevel();
        rf12_sendStart(0x02,teste, (sizeof teste)+1);
      }

     PCintPort::detachInterrupt(1);
    return teste[0];
}

int newLightLevel512(int channel, int mode){
```

```
 interrupts();

  flag=1;
  int level=0;
PCintPort::attachInterrupt(1,flagParty, FALLING);

  while(flag==1)
    {
      level=lightLevel();

      DmxSimple.usePin(DmxPin);
      DmxSimple.maxChannel(maxChannels);
      DmxSimple.write(channel, level);
      DmxSimple.write(channel+128,mode);
    }

     PCintPort::detachInterrupt(1);
    return level;
}
```

int lightLevel(){

```
  return map(analogRead(5), 0, 1023, 0 , 255);
}
```

void flagParty() {

```
  flag=0;
}
```

void start(){

```
  if(firstTime())
    for(int i =1; i  <129; i++)
      {
        noInterrupts();
          EEPROM.write(i,overide);
          DmxSimple.usePin(DmxPin);
          DmxSimple.maxChannel(maxChannels);
          DmxSimple.write(i, 127);
```

```
      DmxSimple.write(i+128, overide);
     interrupts();

     if(i==128)
      EEPROM.write(0,FALSE);
    }
 else
   for(int i =1; i < 129; i++)
    {
     DmxSimple.usePin(DmxPin);
     DmxSimple.maxChannel(maxChannels);
     DmxSimple.write(i+128, EEPROM.read(i));
    }
}
```

boolean firstTime(){

```
 if(EEPROM.read(0)==FALSE)
   return FALSE;
   else
   return TRUE;
}
```

int readChannel(){

```
 // config port as input
 pinMode(DDRC, INPUT);
 pinMode(8, INPUT);
 pinMode(9, INPUT);
 //read adress
 int low =PINC&0x1F;
 int address =((digitalRead(9)<<5)  + low);

 return address+1;
}

int readMode(){
 // config port as input
 Serial.begin(9600);
 pinMode(5, INPUT);
 pinMode(6, INPUT);
```

```
  pinMode(7, INPUT);
  //read Mode
  int mode =(digitalRead(5)<<2) + (digitalRead(6)<<1) + digitalRead(7);
  return mode;
}

void ledBlinking(int temp){
  digitalWrite(led, LOW);   // sets the LED on
  delay(temp);                  // waits for a second
  digitalWrite(led, HIGH);    // sets the LED off
   delay(temp);
  digitalWrite(led, LOW);   // sets the LED on
   }
```

Distributed Wireless Slave Board:

```
//----Libraries----//
#include <JeeLib.h>
#include <Ports.h>
#include <PortsBMP085.h>
#include <PortsLCD.h>
#include <PortsSHT11.h>
#include <RF12.h>
#include <RF12sio.h>
#include <TimerOne.h>
#include <EEPROM.h>

//----Variables----//
int overide = B00;
int newThreshold = B01;
int standAlone = B10;
int RESET = B11;
int ant=0;
int channel=readChannel();
int actualMode;
int actualValue;
int savedPWM=255;
int antValue=512;
int antMode=B1100;
```

```
int minute = 7;
int light;
int desirableLight;
char turnON[] = "1";
char turnOFF[] = "0";

 MilliTimer timer;

//----setup----//
void setup() {
rf12_initialize(2, RF12_868MHZ,channel); //nodeID,HZ,NetGroup
  Serial.begin(57600);
Serial.println(channel);
Timer1.initialize(50);         // initialize timer1, and set timmer @ 20KHz

pinMode(3, OUTPUT);
pinMode(4, OUTPUT);
pinMode(6, OUTPUT);

}
void loop() {
listening();
}

void listening() {
   if (rf12_recvDone() && rf12_crc == 0)
    {
      if ((rf12_hdr&0x1F) == B01)
      {
        actualMode=rf12_data[1]&B011;
        actualValue=rf12_data[0];

        if (RF12_WANTS_ACK)
        {
          rf12_sendStart(RF12_ACK_REPLY, 0, 0);
          Serial.println("->Ack");
        }
        if(antMode==B10)
          setupSensor(turnOFF);
```

```
      mainState();
     }

   if ( (rf12_hdr&0x1F) == B11 && actualMode == B10)
    {
        if(rf12_data[0]&B01)
          lightON();

        if((rf12_data[0]&B01)==0)
          standAloneAwaiting();

          if (RF12_WANTS_ACK)
       {
        rf12_sendStart(RF12_ACK_REPLY, 0, 0);
        Serial.println("->Ack");
       }
       }
    }
}
```

void mainState() {

```
digitalWrite(4, actualMode &B01);
digitalWrite(5, actualMode &B10);

  switch (actualMode) {
    case B00:                   //overide
      if(actualValue != antValue)
      {
        noInterrupts();
          digitalWrite(6, HIGH);  //ENABLE
          analogWrite(9,map(actualValue,0,255,0,400));
          light=lux();
          EEPROM.write(0, light/100);
          EEPROM.write(1, light%100);
          EEPROM.write(2, actualMode);
          antMode=actualMode;
          antValue=actualValue;
         interrupts();
      }
```

```
    break;

    case B01:                 //newThreshold
      if(actualValue != antValue)
      {
        noInterrupts();
          digitalWrite(6, HIGH);  //ENABLE
         analogWrite(9,map(actualValue,0,255,0,400));
          desirableLight=lux();
          EEPROM.write(3, desirableLight/100);
          EEPROM.write(4, desirableLight%100);
          EEPROM.write(2, B10);
          antMode=actualMode;
          antValue=actualValue;
        interrupts();
      }
    break;

case B10:                    //standAlone
      if(actualMode != antMode)
      {
      noInterrupts();
        EEPROM.write(2, B10);
        desirableLight= EEPROM.read(3)*100;
        desirableLight= desirableLight+EEPROM.read(4);
        antMode=actualMode;
      interrupts();
      setupSensor(turnON);
      standAloneAwaiting();
      }
    break;

case B11:                    //SHUTDOWN
      if(actualMode != antMode)
      {
        noInterrupts();

          EEPROM.write(2, B11);
          digitalWrite(6, LOW);  //ENABLE
```

```
        antMode=actualMode;
      interrupts();
    }
  break;
}  //end switch case
}  //end newSetting
```

void standAloneAwaiting(){

```
 //interrupts();
 analogWrite(9,410);
 while(actualMode==B10)
  {
   listening();
  }
}
```

void lightON() {

```
 while(actualMode==B10)
   {
    lightControl();
    listening();
   }
}
```

void standAloneMode() {

```
 while(actualMode==B10)
 {
 //  readTemp();
   lightControl();
   listening();
 }
}
```

int lux() {

```
 return pow(10,map(analogRead(5),0,675,0,3300)/(float)680);
}
```

```
void lightControl() {
  int trsh=desirableLight/5;
```

```
        digitalWrite(6, HIGH);  //ENABLE
  analogWrite(9,map(savedPWM,0,255,0,400));
  int newPWM=savedPWM;
  int dif=desirableLight - lux();
  while( abs(dif=(desirableLight - lux())) > trsh)
    {
     if(dif <trsh)
     {
      if(newPWM>=254)
        newPWM=255;
      else
        newPWM++;
      // Serial.print('P'); Serial.println(newPWM);
      analogWrite(9,map(newPWM,0,255,0,400));
     }
     else
     {
      if(newPWM<=1)
        newPWM=0;
      else
        newPWM=newPWM-1;

      analogWrite(9,map(newPWM,0,255,0,400));
     }
    }
 savedPWM=newPWM;
 }
```

```
void readTemp() {              //reads temperature
  noInterrupts();
  //int Temp=map(analogRead(A4), 0, 1023, 0, 500)-273.15;
 // Serial.println(Temp);
    while(map(analogRead(A4), 0, 1023, 0, 500)-273.15 > 60)
      analogWrite(9, 400);  //turnedOFF
  interrupts();
}
```

```
int readChannel(){
  pinMode(DDRC, INPUT);    // config port as input
```

```
  pinMode(5, INPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);
  int low =PINC&0x0F;   //read adress
  int address =    ((digitalRead(8)<<6)    +   (digitalRead(7)<<5)
                                                          + (digitalRead(5)<<4) + low);
  return address+1;
}

void firstTime() {
  actualMode=EEPROM.read(2);
    switch (actualMode) {
    case B00:    //OVERIDE
      desirableLight= EEPROM.read(0)*100;
      desirableLight= desirableLight+EEPROM.read(1);
      lightControl();
    break;

    case B01:    //NOVO VALOR
    break;

    case B10:    //STANDALONE
      desirableLight= EEPROM.read(3)*100;
      desirableLight= desirableLight+EEPROM.read(4);
     // DmxRxField[channel+128]=B10;
      standAloneAwaiting();
    break;

    case B11:
      desirableLight=350;
      lightControl();
    break;

    case B100:
  digitalWrite(4,LOW);
    break;
  }
}
```

```
void setupSensor(char * signal){
  interrupts();
  for(int i=0; i<10; i++)
    {
      while (!rf12_canSend())
        rf12_recvDone();

    rf12_sendStart(0x22,signal, (sizeof signal)+1); // ACK Pedido

   timer.set(20);
    while(!timer.poll())
    {
      if (rf12_recvDone() && rf12_crc == 0) {

        i=10;
       break;
      }
    }
    }
}
```

Distributed Wired Slave Board:

```
//----Libraries----//
#include <TimerOne.h>
#include <EEPROM.h>
#include <FlexiTimer2.h>
//----Variables----//
int overide = B00;
int newThreshold = B01;
int standAlone = B10;
int RESET = B11;
int ant=0;
int channel=readChannel();
int actualMode;
int actualValue;
int savedPWM=255;
int antValue=512;
int antMode=B1100;
```

```
int minute = 7;
int light;
int desirableLight;


volatile uint8_t  DmxRxField[255]; //array of DMX vals (raw)
volatile uint16_t DmxAddress;    //start address

enum {IDLE, BREAK, STARTB, STARTADR}; //DMX states

volatile uint8_t gDmxState;

//----setup----//
void setup() {
Serial.begin(250000); //Enable serial reception with a 250k rate
gDmxState= IDLE; // initial state
DmxAddress = 1; // The desired DMX Start Adress
Timer1.initialize(50);        // initialize timer1, and set timmer @ 20KHz

pinMode(5, OUTPUT);
pinMode(4, OUTPUT);
pinMode(6, OUTPUT);
firstTime();
}
void loop() {
}

void mainState() {
actualMode=DmxRxField[channel+128];
actualValue=DmxRxField[channel];
digitalWrite(4, actualMode &B01);
digitalWrite(5, actualMode &B10);

  switch (actualMode) {
    case B00:                    //overide
      if(actualValue != antValue)
      {
        noInterrupts();
          digitalWrite(6, HIGH);
```

```
        FlexiTimer2::stop();
        detachInterrupt(0);
        analogWrite(9,map(actualValue,0,255,0,400));
        light=lux();
        EEPROM.write(0, light/100);
        EEPROM.write(1, light%100);
        EEPROM.write(2, actualMode);
        antMode=actualMode;
        antValue=actualValue;
      interrupts();
    }
  break;

  case B01:           //newThreshold
    if(actualValue != antValue)
    {
      noInterrupts();
        digitalWrite(6, HIGH);
        FlexiTimer2::stop();
        detachInterrupt(0);
        analogWrite(9,map(actualValue,0,255,0,400));
        desirableLight=lux();
        EEPROM.write(3, desirableLight/100);
        EEPROM.write(4, desirableLight%100);
        EEPROM.write(2, B10);
        antMode=actualMode;
        antValue=actualValue;
      interrupts();
          }
  break;

case B10:            //standAlone
    if(actualMode != antMode)
    {
    noInterrupts();
      FlexiTimer2::stop();
      detachInterrupt(0);
      EEPROM.write(2, B10);
      desirableLight= EEPROM.read(3)*100;
```

```
        desirableLight= desirableLight+EEPROM.read(4);
        antMode=actualMode;
      interrupts();
      standAloneAwainting();
      }
    break;

case B11:                   //RESET
      if(actualMode != antMode)
      {
        noInterrupts();
          FlexiTimer2::stop();
          detachInterrupt(0);
          EEPROM.write(2, B11);
          analogWrite(9,map(127,0,255,0,400));
          antMode=actualMode;
          detachInterrupt(0);
        interrupts();
      }
    break;

case B100:                  //ShutDown
      if(actualMode != antMode)
      {
        noInterrupts();
          FlexiTimer2::stop();
          detachInterrupt(0);
          EEPROM.write(2, B100);          //overide mode com o valor 0
          antMode=actualMode;
          digitalWrite(4,LOW);
        interrupts();
      }
    break;
}  //end switch case
}  //end newSetting

void standAloneAwainting(){
  interrupts();
  digitalWrite(6,LOW);
```

```
// analogWrite(9,400);
 attachInterrupt(0,standAloneMode, RISING);
 while(DmxRxField[channel+128]==B10);
  }

void standAloneMode() {
 setCounterInt();
 FlexiTimer2::start();
 while(DmxRxField[channel+128]==B10)
 {
   readTemp();
   lightControl();
 }
}

int lux() {
 return pow(10,map(analogRead(5),0,675,0,3300)/(float)680);
}

void lightControl() {
 detachInterrupt(0);

 int trsh=desirableLight/5;

 digitalWrite(6,HIGH);
 analogWrite(9,map(savedPWM,0,255,0,400));
 int newPWM=savedPWM;
 int dif=desirableLight - lux();
 while( abs(dif=(desirableLight - lux())) > trsh)
   {
   // Serial.println(lux());
   // Serial.println(dif);
    if(dif <trsh)
    {
     if(newPWM>=254)
       newPWM=255;
     else
       newPWM++;
    // Serial.print('P'); Serial.println(newPWM);
```

```
     analogWrite(9,map(newPWM,0,255,0,400));
    }
    else
    {
     if(newPWM<=1)
       newPWM=0;
     else
       newPWM=newPWM-1;
    //  Serial.print('P'); Serial.println(newPWM);
     analogWrite(9,map(newPWM,0,255,0,400));
    }
  }
// Serial.println(lux());
 savedPWM=newPWM;

attachInterrupt(0,standAloneMode, RISING);
}
```

void setCounterInt() {

```
  FlexiTimer2::set(10000, standAloneAwainting); // setup interrupt period every 7 min
420000  agora 20sec
}
```

ISR(USART_RX_vect) {

```
  noInterrupts();
     static  uint16_t DmxCount;
     uint8_t  USARTstate= UCSR0A;
     uint8_t  DmxByte   = UDR0;
     uint8_t  DmxState  = gDmxState;

     if (USARTstate &(1<<FE0))
     {
            DmxCount =  DmxAddress;
            gDmxState= BREAK;
     }

     else if (DmxState == BREAK)
     {
            if (DmxByte == 0) gDmxState= STARTB
```

```
            else                    gDmxState= IDLE;
        }
         else if (DmxState == STARTB)
        {
            if (--DmxCount == 0) {
                DmxCount= 1;
                DmxRxField[0]= DmxByte;
                gDmxState= STARTADR;
            }
        }

        else if (DmxState == STARTADR)
        {
            DmxRxField[DmxCount++]= DmxByte;
            if (DmxCount >= sizeof(DmxRxField))
            {
                gDmxState= IDLE;
            }
        }
 mainState();
interrupts();
}
```

```
void readTemp() { // reads temperature
  noInterrupts();
    while(map(analogRead(A4), 0, 1023, 0, 500)-273.15 > 60)
    analogWrite(9, 400);  //turnedOFF
  interrupts();
}

int readChannel(){
  // config port as input
  pinMode(DDRC, INPUT);  //analog pins
  pinMode(5, INPUT);
  pinMode(8, INPUT);
  pinMode(9, INPUT);
  //read adress
  int low =PINC&0x0F;
```

```
  int address =((digitalRead(8)<<6) + (digitalRead(7)<<5) + (digitalRead(5)<<4) +
low);
  //Serial.println(address,BIN); //DEBUG
  //Serial.println(address+1,DEC); //DEBUG
  return address;
}

void firstTime() {
  actualMode=EEPROM.read(2);

 // Serial.println(actualMode);
  switch (actualMode) {
    case B00:    //OVERIDE
      desirableLight= EEPROM.read(0)*100;
    //  Serial.println(desirableLight);
      desirableLight= desirableLight+EEPROM.read(1);
    //  Serial.println(desirableLight);
      lightControl();
    break;

    case B01:    //NOVO VALOR
    break;

    case B10:    //STANDALONE
      desirableLight= EEPROM.read(3)*100;
      desirableLight= desirableLight+EEPROM.read(4);
      DmxRxField[channel+128]=B10;
      standAloneAwainting();
    break;

    case B11:
      desirableLight=350;
      lightControl();
    break;

    case B100:
  digitalWrite(4,LOW);
    break;
```

```
        }
}
```

Wireless Motion Sensor:

```
//----Libraries----//
#include <PinChangeInt.h>
#include <JeeLib.h>
#include <FlexiTimer2.h>

//----Variables----//
char off[] = "0";
char on[] = "1";
boolean TRUE = 1;

MilliTimer timer;

void setup () {
   Serial.begin(57600);
   Serial.println(57600);
   Serial.println("Envia");
   rf12_initialize(3, RF12_868MHZ, 1);
  }

void loop () {
setupSensor();
}

void setupSensor() {
  interrupts();
      if(rf12_recvDone() && rf12_crc == 0 && (rf12_hdr&0x1F) == B10)
        {
          int signal=rf12_data[0]&B1;
          Serial.println(signal,BIN);
            if(signal==B01)
             {
              pinMode(7, INPUT);
              PCintPort::attachInterrupt(7,firstDet, RISING);
             }
```

```
            if(signal==B00)
             {
               PCintPort::detachInterrupt(7);
               FlexiTimer2::stop();
             }

       if (RF12_WANTS_ACK)
        {
          rf12_sendStart(RF12_ACK_REPLY, 0, 0);
          Serial.println("->Ack");
        }
         }
}
```

void firstDet() {

```
PCintPort::detachInterrupt(7);
  Serial.println("Entrei 1");
    envia(on);
    setCounterInt();
    FlexiTimer2::start();
  PCintPort::attachInterrupt(7,secondDet,RISING);
}
```

void secondDet() {

```
  setCounterInt();
  FlexiTimer2::start();
}
```

void envia(char * teste) {

```
        interrupts();
         for(int i=0; i<10; i++)
      {
        while (!rf12_canSend())
          rf12_recvDone();

      rf12_sendStart(0x23,teste, (sizeof teste)+1); // ACK Pedido

    timer.set(20);
      while(!timer.poll())
```

```
      {
        if (rf12_recvDone() && rf12_crc == 0) {
          Serial.println("recebi ACK");
          i=10;
         break;
        }
      }
      }
}
```

void setCounterInt() {

```
 FlexiTimer2::set(10000, turnOFF); // setup interrupt period
}
```

void turnOFF(){

```
PCintPort::detachInterrupt(7);
  envia(off);
  FlexiTimer2::stop();
    PCintPort::attachInterrupt(7,firstDet, RISING);
}
```